NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

PULSE MODE LIGHT SENSING USING FOUR-LAYER SEMICONDUCTOR STRUCTURES AND THEIR APPLICATION IN NEURAL NETWORKS

by

Ioannis Siganos

December 2008

Thesis Advisor:	Gamani Karunasiri
Thesis Co-Advisor:	Murali Tummala

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.			
1. AGENCY USE ONLY ***(Leave blank)***	**2. REPORT DATE** December 2008	**3. REPORT TYPE AND DATES COVERED** Master's Thesis	
4. TITLE AND SUBTITLE: Pulse Mode Light Sensing Using Four-Layer Semiconductor Structures and Their Application in Neural Networks			**5. FUNDING NUMBERS**
6. AUTHOR(S) Ioannis Siganos			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for Public Release; Distribution is Unlimited			**12b. DISTRIBUTION CODE** A

13. ABSTRACT ***(maximum 200 words)***

Existing semiconductor photodetectors produce a steady current or voltage output in response to incident light that depends on the intensity of the light beam. In contrast, biological vision systems produce a stream of pulses with pulse rate representing the amount of incident light power. The goal of this thesis is to explore the use of a four- layer PNPN semiconductor structure as an optical detector that produces pulses instead of steady current or voltage output. The first task of this thesis is to show that the pulse interval distribution is dependent on the intensity of the incident light beam, and that the distribution of pulse intervals is described by a renewal process statistical model. The second task of this thesis is to use the unique characteristics of the pulse generating circuit along with a neural network to construct a 2D sensor array capable of recognizing visual patterns, and thus modeling, to some extent, the human visual cortex.

14. SUBJECT TERMS Thyristor, SCR, Pulse Mode Operation, Poisson Process, Renewal Process, Neural Networks, Back Propagation, Pattern Recognition.			**15. NUMBER OF PAGES** 91
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UU

NSN 7540-01-280-5500 Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

PULSE MODE LIGHT SENSING USING FOUR-LAYER SEMICONDUCTOR STRUCTURES AND THEIR APPLICATION IN ARTIFICIAL NEURAL NETWORKS

Ioannis Siganos
Lieutenant, Hellenic Navy
B.S., Hellenic Naval Academy, 1997

Submitted in partial fulfillment of the
requirements for the degrees of

MASTER OF SCIENCE IN APPLIED PHYSICS

and

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 2008

Author: Ioannis Siganos

Approved by: Gamani Karunasiri
Thesis Advisor

Murali Tummala
Thesis Co-Advisor

James H. Luscombe
Chairman, Department of Physics

Jeffrey B. Knorr
Chairman, Department of Electrical and Computer Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Existing semiconductor photodetectors produce a steady current or voltage output in response to incident light that depends on the intensity of the light beam. In contrast, biological vision systems produce a stream of pulses with pulse rate representing the amount of incident light power. The goal of this thesis is to explore the use of a four-layer PNPN semiconductor structure as an optical detector that produces pulses instead of steady current or voltage output. The first task of this thesis is to show that the pulse interval distribution is dependent on the intensity of the incident light beam, and that the distribution of pulse intervals is described by a renewal process statistical model. The second task of this thesis is to use the unique characteristics of the pulse generating circuit along with a neural network to construct a 2D sensor array capable of recognizing visual patterns, and thus modeling, to some extent, the human visual cortex.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

A four-layer (PNPN) semiconductor structure, commonly known as Silicon Control Rectifier (SCR) or thyristor can be used to create a pulse generation circuit by connecting it in series with a parallel resistor and capacitor. This is due to the bias controlled switching characteristics of the SCR. The SCR operational theory, as well as, the origin of pulse generation will be addressed first. They are followed by the procedure used to measure the optical response of a FDS100 silicon photodiode that will be used in the SCR circuit for making a pulse-mode optical sensor.

In this thesis, a detailed experimental study of the spectral dependence of the pulsed rate at room temperature is presented. Monochromatic light will be incident on a silicon photodiode, connected in the reversed bias configuration to an SCR circuit at constant bias. The pulses generated under light illumination will be captured using an A/D converter and the pulse rate along with all the necessary information for the statistical analysis will be extracted from the file produced by a LABVIEW data acquisition program. It will be shown, that the pulse rate follows the photocurrent response of the Si photodiode, thus enabling the use of the circuit as a pulse-mode optical detector. The pulse mode detection is not limited to Si photodiode's spectral range but can be extended to other wavelengths by replacing it with photodiodes made of other semiconductors, for example in the IR region. Also, the SCR circuit can be used as an optical detector without the use of the external Si photodiode, by directly shining light into the middle p-n junction of the SCR. In addition, the inter-pulse interval can be characterized by a renewal process statistical model. Finally, the produced pulses have similar characteristics (periodicity, refractory period, and burstiness) as the spikes produced by the biological neurons to the presence of external stimuli, thus, enabling the detectors to potentially be used as pixels for fabrication of an artificial vision system similar to that of a biological vision system.

The brain is a highly complex, nonlinear, and parallel computer. It has the capability to organize its structural constituents, known as neurons, so as to perform certain computations, (e.g., pattern recognition and motion control), many times faster

than the fastest digital computer in existence today. Computers cannot solve all the real-world applications due to their sequential processing. Thus, features from physiology of the brain are used as the basis of the processing models, known as neural networks. The basic concepts of the neural networks will be presented, the neuron model and the neural networks architecture. In addition, the learning process will also be addressed. It is a property that is of primary significance for a neural network, because it provides the network with the ability to learn from its environment and to improve its performance through learning, using a prescribed measure. A neural network learns about its environment through an interactive process of adjustments applied to its synaptic weights and bias levels. Finally, the back propagation algorithm will be described.

The retina's task is to convert an optical image into a neural image for transmission down the optic nerve for further analysis. The processing in the primary visual cortex consists mainly of an orientation response; that is a response determined by the directions of lines and edges in the visual image. This is how humans perceive the shape of an object. When an object enters the visual field, the neurons sensitive to the orientation of the edges of the object will respond by firing a number of spikes.

A 2D 3×5 sensor array consisting of fifteen of the photodiode-SCR circuits and a Wildcat BL2000 microprocessor will be constructed. The numerical characters from 0 to 9 will be projected on the array of the photodiodes using a light source. The photodiodes of the sensor array that are illuminated produce photocurrent, causing the corresponding SCR circuits to produce a series of pulses. Hence, the photodiode-SCR circuits will mimic the retina's behavior. A neural network will be trained to solve this pattern recognition problem using the Neural Networks Toolbox in MATLAB. After the neural network is trained, its parameters (synaptic weights and biases) will be extracted and transformed to a simple program in Dynamic C for implementation in the Wildcat BL2000. The Wildcat BL2000 microprocessor will transform the image of the numerical character into a 15-bit number and then it will output the corresponding numerical character using the above program. Thus, the constructed sensor models, to some extent, the human vision cortex. Finally, two robots will be activated by the sensing circuitry depending on the recognized number.

ACKNOWLEDGMENTS

The author would like to acknowledge the following faculty and staff of the Naval Postgraduate School:

1. Professor G. Karunasiri, for his guidance throughout the course of this research, even during the ECE part of it. His knowledge and intuition was invaluable and working under his supervision is an experience I will always treasure.

2. Professor M. Tummala, for his ideas, and guidance that made this thesis very interesting. Specifically, it was his idea to construct a sensor using the SCR circuit, implementing a neural network algorithm to make it recognize the numerical characters, and thus modeling the human visual cortex.

3. Professor R. Harkins. Although I was never one of his students, he agreed to help me, gave me access to his lab and solved two of the problems I have encountered during the construction of the sensor.

4. Sam Barone, the Physics Department's Electronic Technician, for his guidance, patience, and assistance, especially during the construction of the sensor. His "can-do" attitude gave me hope and I always knew that I could count on him.

5. Jay Adeff, for writing the LABVIEW script to capture the pulses.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

Existing semiconductor photodetectors produce a steady current or voltage output in response to incident light that depends on the intensity of the light beam. In contrast, biological vision systems produce a stream of pulses with pulse rate representing the amount of incident light power. The goal of this project is to explore the use of a four-layer PNPN semiconductor structure as an optical detector that produces pulses instead of steady current or voltage output. The first task of this thesis is to show that the pulse interval distribution is dependent on the intensity of the incident light beam, and that the distribution of pulse intervals is characterized by a renewal process statistical model. The second task is to use the unique characteristics of the pulse generating circuit and a neural networks model to construct a 2D sensor array capable of recognizing visual patterns.

This chapter gives a brief introduction to the physics of pulse generation using a PNPN device and outlines the related work on current concepts for visual prostheses, as well as, some previous experimental evidence of the aperiodic nature of the pulse intervals. This is followed by a description of the data collection method used to measure the optical response of a FDS100 silicon photodiode that will be used in the SCR circuit for making a pulse-mode optical sensor. The second chapter of the thesis describes the observed pulse rate dependence on the intensity of the incident illumination. This is followed by an analysis of this behavior using a statistical model. The third chapter presents the neural networks model. In the fourth chapter a 2D optical sensor array capable of recognizing visual patterns using the unique characteristics of the thyristor circuit and a neural network will be presented. The final chapter summarizes the results of this thesis and discusses possible areas of future research.

A. THYRISTOR OPERATIONAL THEORY

Spontaneous generation of voltage pulses was observed in a circuit (see Figure 1) that consists of a Silicon Controlled Rectifier (SCR) in series with a parallel RC circuit under a DC bias. The pulse rate was found to increase with the DC bias and saturated as the pulse period reached the RC time constant. Shining light directly on the SCR middle

p-n junction or injection of current into it using a photodiode connected to the gate terminal was also found to increase the pulse rate with light intensity [1].

In order to understand the way the pulses are produced using the circuit in Figure 1, the SCR operational theory will be addressed first.

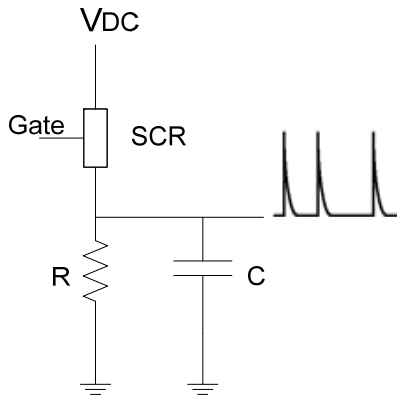


Figure 1. SCR Circuit Schematic

1. I-V Characteristic

SCRs have a current versus voltage or I-V characteristic curve, which is represented in Figure 2.

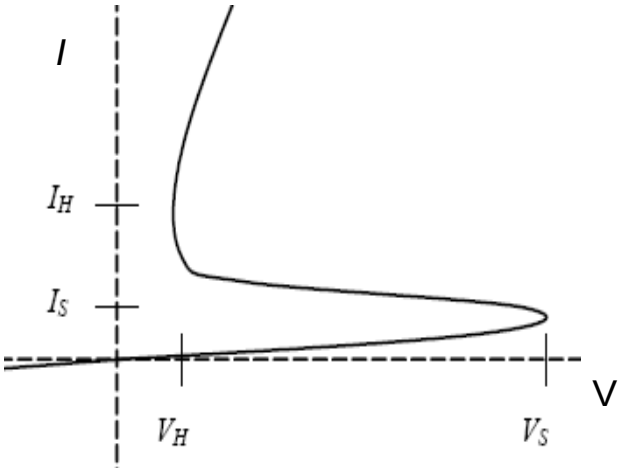


Figure 2. Typical Current-Voltage Curve of a SCR (After [2]).

The I-V curve can be broken into several operating modes. The focus of this section is limited to the forward-biased region since that is the region of interest in optical detection with thyristors.

Beginning from zero, increasing the bias voltage initially produces only a very slowly increasing anode current. When the anode current is less than the switching current I_S, the operating mode is called the forward blocking mode. It is characterized by very high resistance to current. The switching voltage V_S is the maximum forward voltage across the device. When the current exceeds I_S the device enters the negative resistance mode. As the name indicates, this mode is characterized by negative resistance where increasing anode current results in anode-to-cathode voltage lower than V_S. Once the anode current exceeds the holding current I_H, the device is considered to be ON. This is the forward conducting mode and it is characterized by extremely low resistance [2].

As the DC bias across the SCR approaches the switching voltage, the injection of electrons and holes into the middle reverse biased p-n junction by the two outer forward biased p-n junctions rapidly increases as illustrated by the band diagram in Figure 3. These injected carriers sweep across the middle p-n junction and accumulate in the potential wells formed near the outer two p-n junctions as shown by the band diagram of the SCR. These accumulated charges tend to attract additional charge carriers with opposite polarity from the two outer layers and the process quickly becomes regenerative. The accumulation of charges generates an increasing electric field in the middle p-n junction opposite to that of the applied bias causing it to forward bias, thereby sending the SCR to the ON state [1].

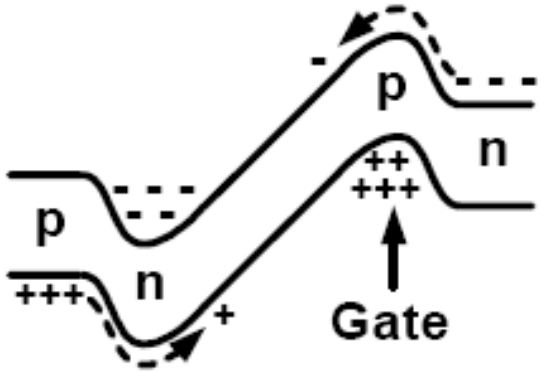


Figure 3. Schematic Band Diagram and Charge Accumulation in the SCR just below the Switching Voltage (in the OFF state). (After [1]).

It is important to emphasize the distinction between the switching point and the holding point. Switching is not synonymous with turning ON. The device begins turning ON at the switching point, but is not fully ON until the holding point is reached. SCRs are usually designed to enhance the turning-ON process and do not normally operate between the switching and holding points [3].

2. Switching Mechanisms

There are many ways to switch a SCR ON. The first one is increasing the bias voltage until the anode current exceeds the switching current. As long as the biasing source is not current-limited below the holding current, the device will switch ON. This mechanism was further investigated by Matos [4].

Thyristors with a gate contact (e.g., SCRs) can be switched using an appropriate gate current, which is usually much smaller than the anode current. This method is typically used in the classic application of the thyristor as a switch. Increasing the gate current results in a decrease in the switching voltage V_S [3]. The injected current via the gate terminal increases the accumulation of positive charge (see Figure 3) which accelerates the regenerative process causing the thyristor to switch faster at a lower bias [1], as depicted in Figure 4.

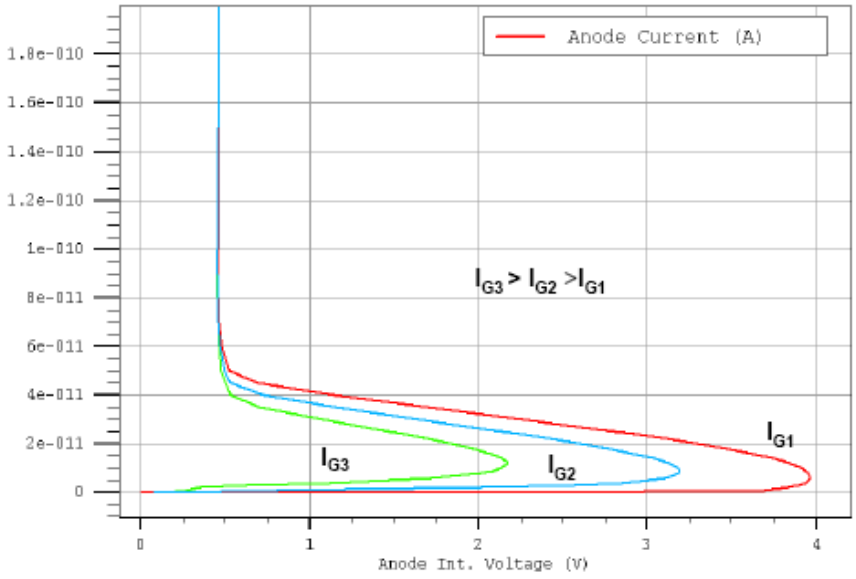


Figure 4. Reduction in V_S with increasing I_G. (From [3]).

Thyristors can also be switched by using incident light, which is incident either on an open thyristor or on a photodiode, with its cathode connected at the gate of the thyristor and its anode to the ground (reversed bias configuration). Increasing incident light intensity acts much like increasing gate current to lower the switching point, except the carriers are much more quickly swept through the depletion region, since the photocurrent adds to the hole current [5]. This switching mechanism will be the main focus of this thesis and will be explored further.

3. Origin of Pulse Generation

In order to properly bias the SCR circuit for sustained pulsing operation, the values for the capacitor and the resistor must be carefully selected. It was found that the capacitance must be chosen to be large enough to provide a low impedance path (~1 Ω) during the switching, given the switching rise time (τ = 200 nsec) and the peak transient current (CV_p/τ) of the SCR, where V_p is the pulse height. The resistor must be chosen to be small enough to maintain the current needed for switching and large enough to keep the current below the holding current of the SCR [1].

During the switching of SCR, the current primarily flows through the capacitor, due to its lower impedance, causing it to charge rapidly. Since the voltage across the capacitor increases due to charging, the voltage across the SCR decreases forcing it back to OFF state. Now, that the thyristor is in OFF state, the capacitor drains off its charge through the resistor, and thus decreasing its voltage. When enough charge is depleted, the thyristor can switch ON again, if the voltage across it exceeds the threshold voltage, V_S. The process repeats itself, generating series of pulses.

During all the measurements conducted throughout this thesis, a Motorola MBS 4991 Silicon Bilateral Switch (SBS) was connected in series with a parallel RC circuit having values of 33 kΩ and 27 nF, respectively. The RC time constant was calculated to be 0.89 msec.

B. SUMMARY OF PREVIOUS EXPERIMENTAL RESULTS

Most current concepts for visual prostheses are based on neuronal electrical stimulation of different locations of the visual pathways within the central nervous system [6-18]. For example, an array of microelectrodes [7, 8] has been developed that can be implanted in the eye to stimulate optical nerves. The electrode array was connected using thin wires to an extra ocular unit that contained electronics for wireless data, power, and generation of stimulus current which was implanted behind the ear [9]. The electrodes were embedded in a silicone-based material to achieve biocompatibility. The visual signals from a video camera that were processed through a microcomputer and power to the extra ocular unit were provided via a 1 MHz inductive wireless link using an external antenna. The array had only 4×4 pixels. These devices were implanted in several patients who were able see shapes and outlines and pour a liquid from one cup to another. An integrated approach produced an implantable artificial retina containing several thousand small photodetectors [10], each attached to a miniature electrode on the back of the chip. The light falling on the detectors were supposed to activate the electrodes, stimulating the ganglion cells on the surface of the retina. However, the visual sensation was found to occur in retinal areas distant from the implant [10]. Since the current generated under normal light conditions by such photodetectors was not sufficient to generate a visual sensation in the brain, a near infrared light beam was employed to increase the current [13]. In addition, a microphotodiode array with integrated signal processing electronics at each pixel to convert the photosignal to a stream of voltage pulses was reported [14], but it was found that the relatively high current consumption per pixel (~85 μA) by the signal processing electronics could limit the use of the microphotodetector array in retinal stimulation. Alternatively, a thin array, containing 64 electrodes implanted inside the skull on the surface of the occipital lobe of the brain [15] has been developed. The device is connected to an electrical socket that passes through the skull and skin. Outside the skull, the device is similar to the retinal stimulators, with a television camera mounted on glasses and a small computer to process the signal. A moderately high voltage is required to stimulate the target cells inside the brain, and it is not possible to stimulate small groups of cells specifically. The voltages applied could, in some cases, provoke epileptic seizures. In order to overcome this

problem, electrode arrays were implanted in the interior of the brain, directly in close proximity to the cells to be stimulated [16-18]. Artificial vision implants to date are thus primarily based on stimulation of either retina or brain with an array of electrodes using signals from an external camera and they have not yet been effective. This project is aimed at using pulses generated by an SCR circuit under light illumination for pattern recognition based on neural network processing.

The observation of interest made by Karunasiri in [1] is that the inter-pulse interval does not depend solely on the RC time constant as expected. Rather, it is intrinsic to the SCR and the strength of the DC bias across the circuit. The inter-pulse interval can be arbitrarily increased by reducing the DC bias across the circuit. At low bias, the rate of injection of minority carriers into the reverse biased middle p-n junction is relatively small which in turn slows the regenerative process and hence increases the time required for generating the electric field necessary for switching the SCR. This is also responsible for the aperiodic nature of the time intervals between pulses. Figure 5 shows typical observed voltage output for the thyristor circuit for high DC bias (periodic) pulses and low DC bias (aperiodic) pulses.

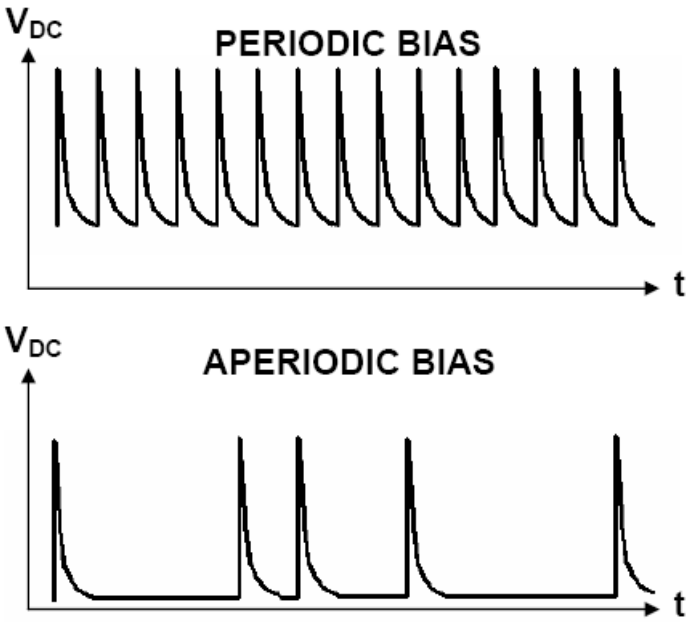


Figure 5. Typical observed sequences of pulses as a function of time or high DC bias (top) and low DC bias (bottom). (From [3]).

The pulse rate prior to saturation (due to the RC time constant of the circuit) is determined by the amount of current injected into the middle PN junction by the two forward biased outer PN junctions. The bias dependence of the injected current can be obtained using the standard diode equation and thus the average frequency dependence on bias is expressed as:

$$\langle f \rangle = f_o e^{\frac{e(V_{SCR}-V_T)}{\eta kT}} \tag{1.1}$$

where η is the ideality factor, f_o is a parameter, and V_T is the threshold voltage for pulsing which depends on the switching voltage of the SCR [1]. By taking the inverse of both sides, the average time interval dependence of the PNPN device as a function of applied bias can be understood as follows:

$$\langle t \rangle = t_o e^{\frac{-e(V_{SCR}-V_T)}{\eta kT}} \tag{1.2}$$

where $t_o = \frac{1}{f_o}$ is related to the RC time constant.

Matos [4] has shown that an increase in temperature lowers the required switching current greatly, and slightly lowers the switching voltage required to turn ON the thyristor. He has also determined that the spontaneous pulse generation observed by changing the voltage bias or temperature, can be classified as a memoryless process, and specifically that can be described by the Poisson point process statistical model. In the following section, the experimental setup used for the spectral characterization of the pulse mode circuit is presented.

C. CONFIGURATION FOR MEASURING OPTICAL RESPONSE

The basic configuration used in most of the optical measurements in this thesis is depicted in Figure 6. This configuration was slightly modified, depending on the task performed.

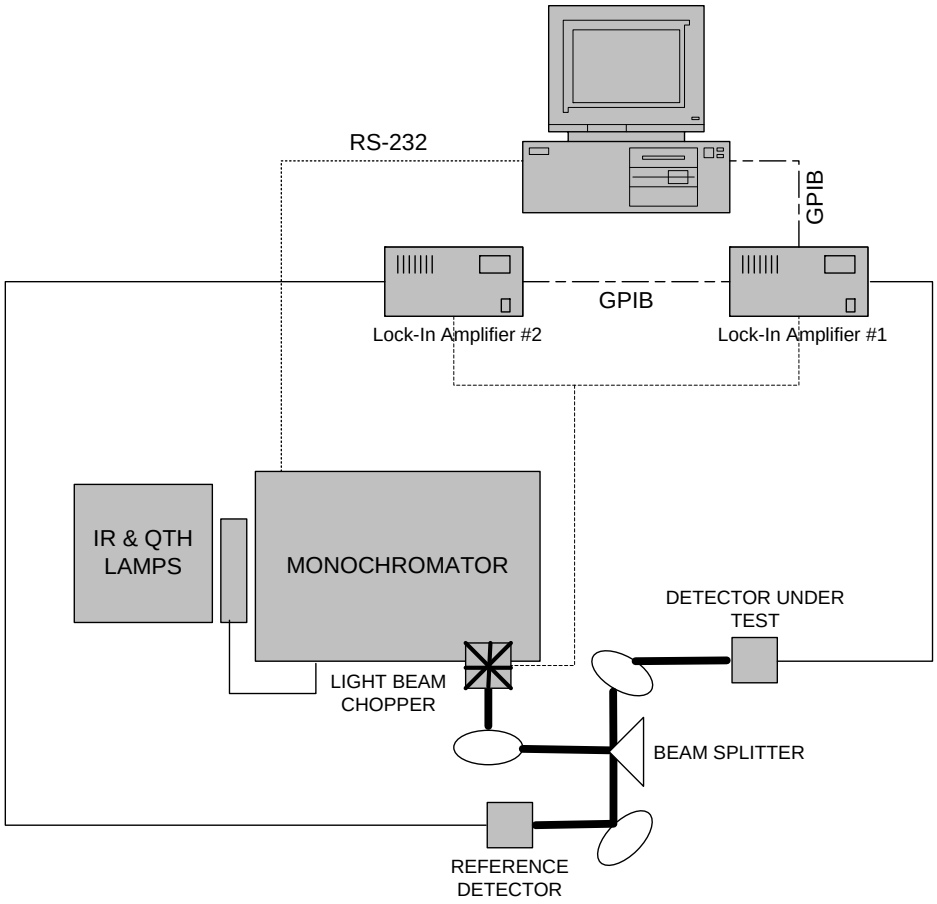


Figure 6. Experimental setup used for measurements.

The system consists of 100-W Quartz Tungsten Halogen (QTH) light source which produces light wavelengths from 200 nm to 3500 nm. The light then enters into the monochromator through a slit (set at 0.5 mm for the resolution needed) and a motorized filter wheel with a set of filters. These filters remove the higher order wavelengths in the output light beam. The motorized filter wheel is automatically controlled by the monochromator.

The monochromator (ORIEL CORNESTONE 260 1/4) is computer controlled via LABVIEW Virtual Instrument script. It uses a stepping motor to drive the three diffraction gratings that project the desired wavelength set by the user on the output slit of the monochromator. The output slit was again set at 0.5 mm to match that of the input.

The light beam (of specific wavelength) that exits the monochromator is chopped by a mechanical chopper. The chopping frequency was set to 140 Hz and was also fed into the lock-in amplifiers.

After the chopping, the beam is guided by a collimating off-axis parabolic mirror, on a beam splitter (90º prism) that split the beam in half. The beam is split so that the response of the test detector could be compared to that of a reference to account for variations in power emitted by the source or absorbed along the optical path.

The two parts of the light beam are guided individually by two different off-axis parabolic mirrors on the test and reference detectors. They are positioned in such a way that the light path to the test detector was equal to the light path to the reference detector. Their focal point is set at the two detectors, respectively.

The photocurrents produced by the two detectors were measured by the two lock-in amplifiers (Model 5210 from EG&G). The signal to be measured was made to appear at a reference frequency of 140 Hz from the light beam chopper. The signal was then amplified and applied to a phase sensitive detector operated at the reference frequency. The result was a detector output that included a value representing the amplitude of the signal of interest as well as higher order harmonics which can be filtered using appropriate filters in the lock-in.

The characterization system in Figure 6 was operated by interfacing with a computer using LABVIEW software. The user set the desired wavelength or a scan of wavelengths to be performed by a LABVIEW Virtual Instrument script. The same script was able to record the output of the two lock-in amplifiers and save to a text file containing the necessary information for further analysis.

1. Prior to the Measurements

One of the most challenging tasks, besides focusing the beams on the two detectors, was to find the exact location of the splitter, where the beam was split in half. This was done using the following procedure.

Two identical Si photodiodes (FDS 100 by THOR LABS) played the role of the test and the reference detectors. Since the path of the beam was the same and two identical photodiodes were used, it is easy to determine that if the splitter was positioned

correctly, the currents measured by the two lock-in amplifiers should be the same. Series of scans were performed, from 200-1100 nm, each scan corresponding to a different position of the splitter. It was actually found that the two Si photodiodes were slightly different. A MATLAB code was written in order to determine the minimum of the mean difference between the two current measurements at each wavelength. At a specific position of the splitter, the minimum mean difference was found to be less than 3.4 nA compared to a signal of 2.42 μA at 980 nm. The splitter was set at this position for the measurement of the responsivity of a silicon photodiode described in the following section.

2. Measuring the Responsivity of a FDS100 Si Photodiode

Responsivity R quantifies the amount of output seen per watt of incident optical power [19]. It is a measure of the sensitivity to light, and it is defined as the ratio of the photocurrent I_p to the incident light power P at a given wavelength:

$$R = \frac{I_p}{P} \tag{1.3}$$

In other words, it is a measure of effectiveness of conversion of the light power into electrical current. It varies with the wavelength of the incident light as well as applied reverse bias and to some extent with operating temperature. According to equation (1.3), in order to determine the responsivity of a detector one should measure the photocurrent of the detector and divide it with the light power incident on it. The former can be measured with the use of the lock-in amplifiers described earlier. As per the latter, it was indirectly measured with the use of another Si photodiode with known responsivity in the position of the reference detector in Figure 6. This was a Si photodiode, Model UV-035D, which was calibrated by UDT SENSORS, INC. Its responsivity as a function of wavelength is depicted in Figure 7.

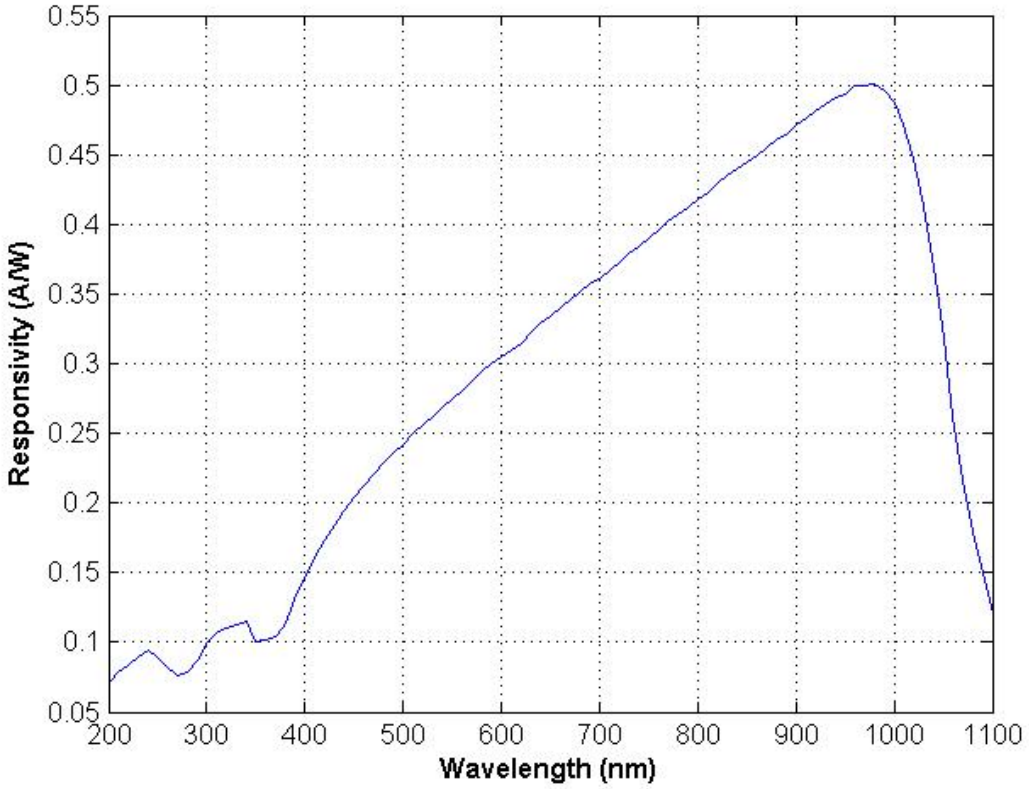


Figure 7. Responsivity vs Wavelength for the calibrated Si photodiode.

The procedure of measuring the responsivity of a test detector is as follows. Since, as described previously, the light path is the same and the light beam was split in half, the intensity of the light beam at the test detector (FDS100 Si photodiode) is same as that at the reference Si photodiode, that is:

$$\mathrm{I}_t = \mathrm{I}_r \tag{1.4}$$

where the subscripts "t" and "r" denote the test and reference detector, respectively. But the intensity is given by the ratio of the incident power to the area of the detector, so equation (1.4) gives:

$$\frac{P_t}{A_t} = \frac{P_r}{A_r} \tag{1.5}$$

Solving (1.3) for power and substituting in (1.5), we get :

$$\frac{I_t}{R_t \cdot A_t} = \frac{I_r}{R_r \cdot A_r} \tag{1.6}$$

and finally, solving for R_t we get :

$$R_t = R_r \cdot \frac{I_t}{I_r} \cdot \frac{A_r}{A_t} \tag{1.7}$$

Equation (1.7) enables us to compute the responsivity of a detector of interest without having to measure the absolute power incident on it. It must be noted, though, that the reference detector must have broader spectral characteristics than the test detector. For example, the responsivity of a Ge photodiode cannot be estimated using the calibrated Si photodiode, since its response extends beyond the cutoff of wavelength of Si (1200 nm) [20].

For testing the performance of the measurement system, the two photocurrents I_t and I_r generated by the two Si photodetectors were measured using the two lock-in amplifiers. The areas of the test and reference detector, A_t and A_r, were found from the corresponding photodiode specifications and the width of the slit at the monochromator, which was set at 0.5 mm. Using $A_t = 3.6\,\text{mm} \cdot 0.5\,\text{mm} = 1.8\,\text{mm}^2$ and $A_r = 5.8\,\text{mm} \cdot 0.5\,\text{mm} = 2.9\,\text{mm}^2$, a MATLAB code was written to compute and plot the responsivity. The data were extracted from the text file produced by the LABVIEW Virtual Instrument script.

The plot of the responsivity of the FDS100 Si photodiode is depicted in Figure 8 (red dashed line). It is plotted against the responsivity of the calibrated Si photodiode (blue line) and the responsivity of an FDS100 Si photodiode (red crosses) as can be found in [20]. The slight difference between the standard response curve provided by the manufacturer could be due to process variations for different batch of diodes as well as the use of a different reference photodiode.

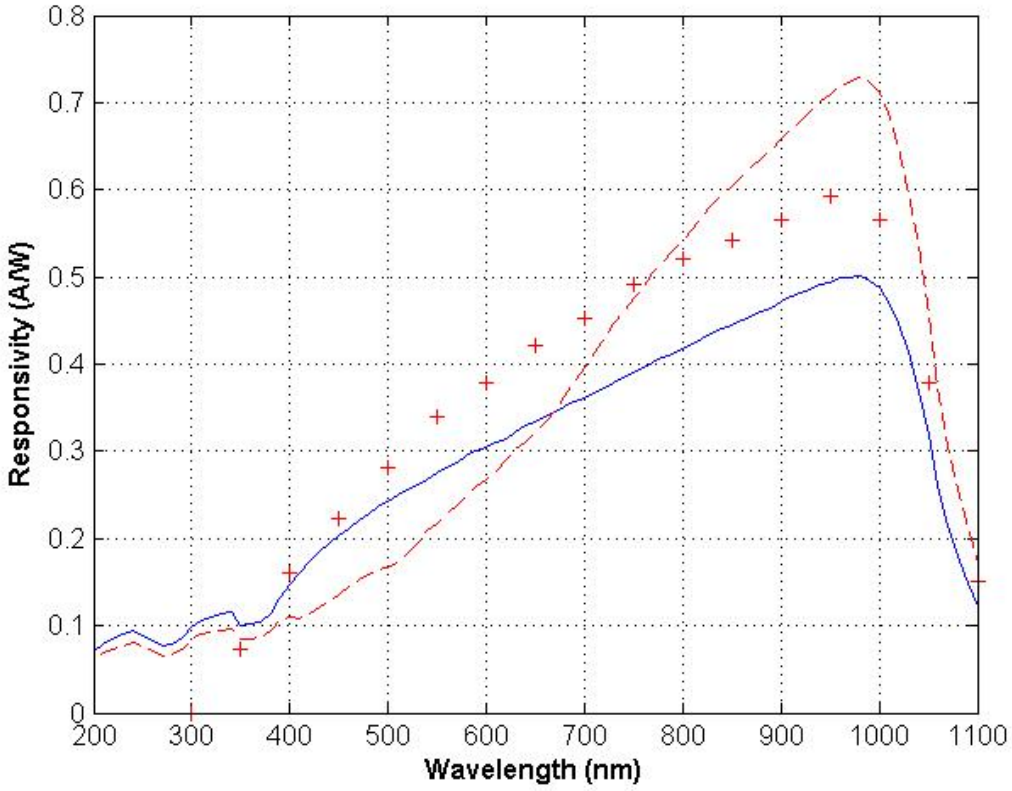


Figure 8. Measured Responsivity for a FDS100 Si photodiode (red dashed line). It is plotted against the responsivity of the calibrated Si photodiode (blue line) and the Responsivity of an FDS100 Si photodiode (red crosses), as can be found in the THOR LABS specifications.

D. SUMMARY

This chapter gave a brief introduction to the operational theory of the SCR, its switching mechanisms, as well as, the physics of pulse generation using a SCR circuit. It also outlined the related work on current concepts for visual prostheses, as well as, some previous experimental evidence of the aperiodic nature of the pulse intervals. Finally, the configuration and the procedure used to measure the responsivity of a FDS100 Si photodiode were presented.

The following chapter will describe the observed pulse rate dependence on the wavelength of the incident illumination and the statistical model that will be used to analyze this behavior.

II. THYRISTOR CIRCUIT CHARACTERISTICS

This chapter discusses the observed pulse rate dependence on the intensity of the incident monochromatic illumination on a FDS100 Si photodiode that is connected to an SCR circuit. The first goal is to show that the above circuit can be used to construct pulse-mode optical sensors. It is followed by an analysis of this behavior using a renewal process statistical model. The second goal is to prove that the spontaneous pulse generation observed can be classified as a memoryless process similar to that observed in action potentials of biological neurons.

A. MEASURING THE PULSE RATE

1. Measuring the Absolute Power

In order to measure the pulse rate, the absolute power should be first determined. This was done using the configuration depicted in Figure 6. Again, a UV-035D Si photodiode with known responsivity was used, this time at the position of the test detector. The slit width of the monochromator was set to 0.5 mm. The splitter was positioned in such a way that it would act as a mirror and guide the beam to the test detector. This time the power was maximum at the position of the test detector.

Since the photodiode has a known responsivity, and the photocurrent is measured by a lock-in amplifier, the absolute power can be calculated from equation (1.3) as follows:

$$R = \frac{I_p}{P} \Rightarrow P = \frac{I_p}{R} \tag{2.1}$$

The calculated power for the UV-035D Si photodiode was then corrected for a FDS100 Si photodiode, using equation (1.5):

$$\frac{P_{FDS100}}{A_{FDS100}} = \frac{P_{UV35D}}{A_{UV35D}} \Rightarrow P_{FDS100} = P_{UV35D}\frac{A_{FDS100}}{A_{UV35D}} \tag{2.2}$$

where $A_t = 3.6\,\text{mm} \cdot 0.5\,\text{mm} = 1.8\,\text{mm}^2$ and $A_r = 5.8\,\text{mm} \cdot 0.5\,\text{mm} = 2.9\,\text{mm}^2$.

A MATLAB script was used to produce the plot of the absolute power as a function of wavelength as shown in Figure 9.

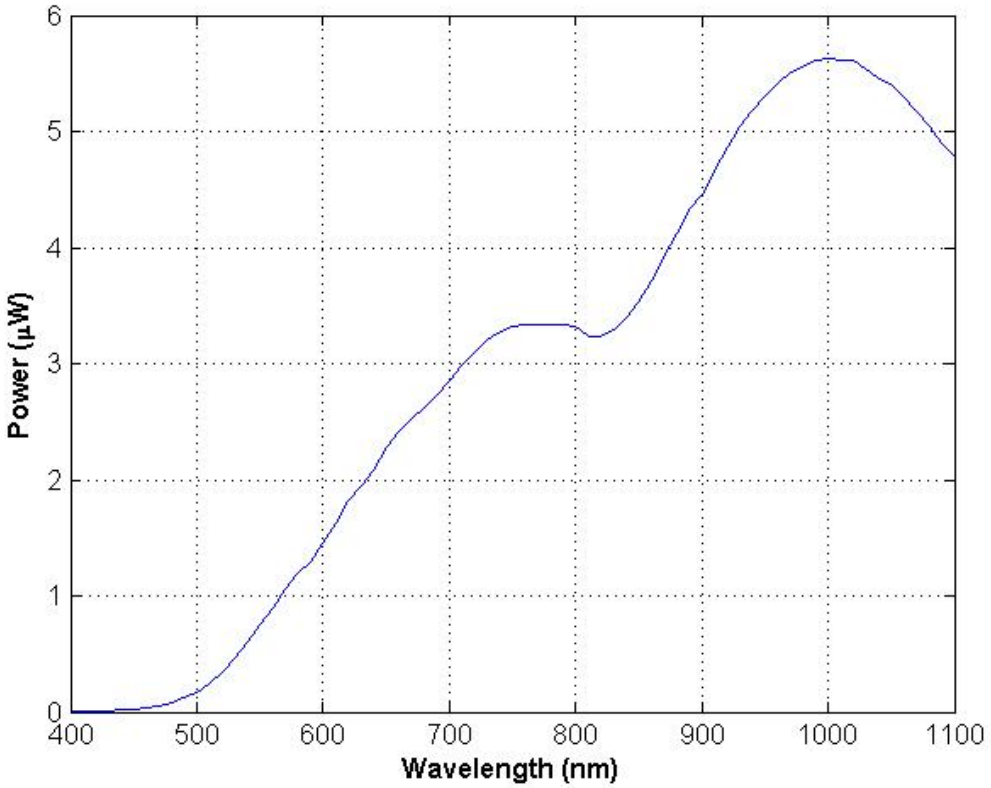


Figure 9. Absolute power of an FDS100 Si photodiode as a function of wavelength.

In the need for more power incident on the test detector, the slit width was changed to 1 mm, but the lock-in amplifier was overloaded when the photocurrent exceeded its scale (3 μA). To overcome this problem, the dependence of the slit width and photocurrent was calculated for specific wavelengths. If for the same wavelength, the ratio of the photocurrent for different slit widths was the same, then the estimate of the absolute power would equal the absolute power already calculated for 0.5 mm times the ratio. The above measurement was done for every 50 nm, starting from 400 to 650 nm. Having the wavelength constant, the slit width was changed every 0.1 mm, from 0.5 to 1 mm, and the photocurrent was measured. It turned out that the ratio was almost the same for the same wavelength. When the slit width was doubled (from 0.5 mm to 1 mm), one should expect that the ratio would be four, but it was actually 4.45. The same happened

when the slit width was changed from 0.5 to 0.8 mm. The ratio should be $(0.8/0.5)^2 = 2.56$, but it was 2.79. This is an indication that the beam is not uniform. Some typical values of the measured power for a slit width of 0.5 mm, and the estimated power for a slit width of 0.8 mm ($P_{0.5} \times 2.79$) can be seen in Table 1.

Wavelength (nm)	**Absolute Power (μW) for slit width at 0.5 mm**	**Estimated Absolute Power (μW) for slit width at 0.8 mm**
400	0.01	0.03
500	0.173	0.48
600	1.451	4.05
700	2.859	7.98
800	3.319	9.26
900	4.461	12.45
1000	5.629	15.71
1100	4.784	13.35

Table 1. Values of the measured power as a function of wavelength

2. Configuration

In order to measure the pulse rate as a function of wavelength, the apparatus depicted in Figure 10 was placed at the position of the test detector. It consists of an FDS100 Si photodiode connected to its SCR circuit, which was inside the metallic box.

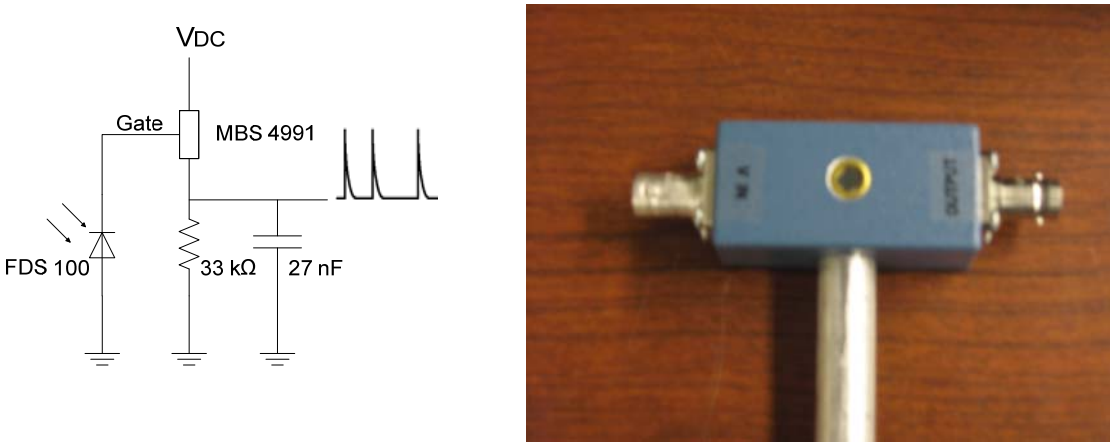


Figure 10. The circuit and apparatus used for measuring the pulse rate.

The experimental set up for measuring the pulse rate can be depicted in Figure 11. This time, monochromatic light should have a constant illumination and that is why there are no light beam chopper and lock-in amplifiers. The light beam was guided to the apparatus of Figure 10 through two parabolic mirrors and one splitter, and it was focused on the FDS100 Si photodiode of the apparatus. The power supply provided the necessary bias for pulsing. The output pulses were observed on the oscilloscope and were counted by a National Instruments USB-6099 board. The above configuration was facilitated by the use of a computer control system which controlled the monochromator as before (see Figure 6). In addition the digitized waveform of the produced pulses could be seen on its monitor. A LABVIEW Virtual Instrument script was used to count the pulses and record their number and time of occurrence in an Excel file for further analysis.

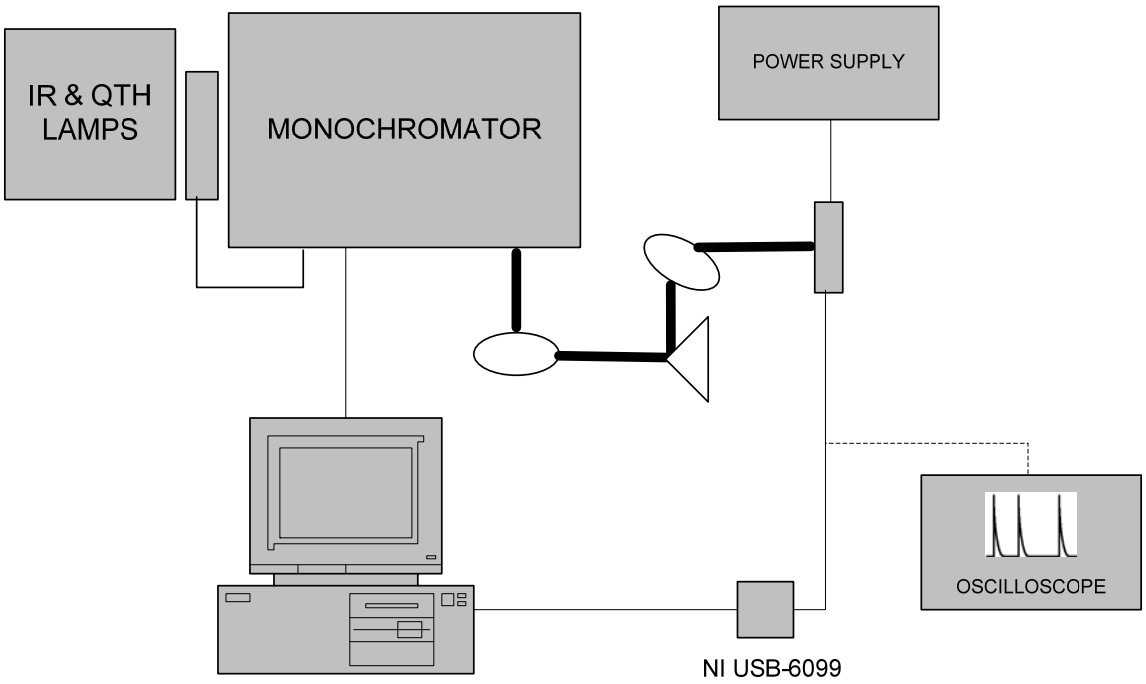


Figure 11. Experimental setup used for the measurement of pulse rate.

During the measurements, the DC bias across the circuit was adjusted such that there were no pulses without light incident on the photodiode. It was found that, at 16.1 V, there were no pulses recorded. This also ensures that the background illumination would not affect the experiment. The slit width was set to 0.5 mm. For every wavelength, ten measurements were taken and the pulse rate was calculated by averaging the number of the captured pulses in one second.

Figure 12 depicts the measured pulse rate as a function of wavelength. It can be seen that up until 620 nm, only a small number of pulses was captured (less than 30).

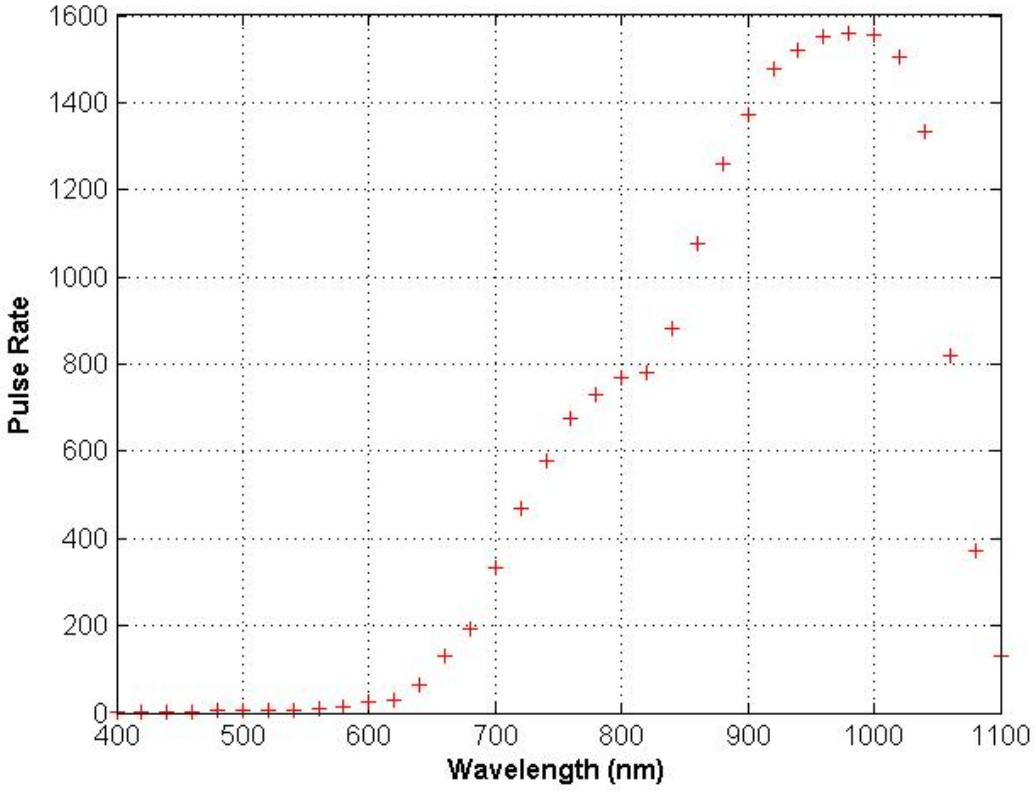


Figure 12. Measured Pulse Rate as a function of Wavelength.

The comparison of the above curve and the raw data of the photocurrent taken during the measurement for the responsivity of a Si FDS 100 photodiode from the previous chapter can be depicted in Figure 13. The plots are normalized (both divided by the corresponding maximum value), so the biggest value is one and it appears at 980 nm for both curves. It is clear that the pulse rate curve follows the photocurrent curve. The slight difference between them could be due to the fact that different Si photodiodes were used.

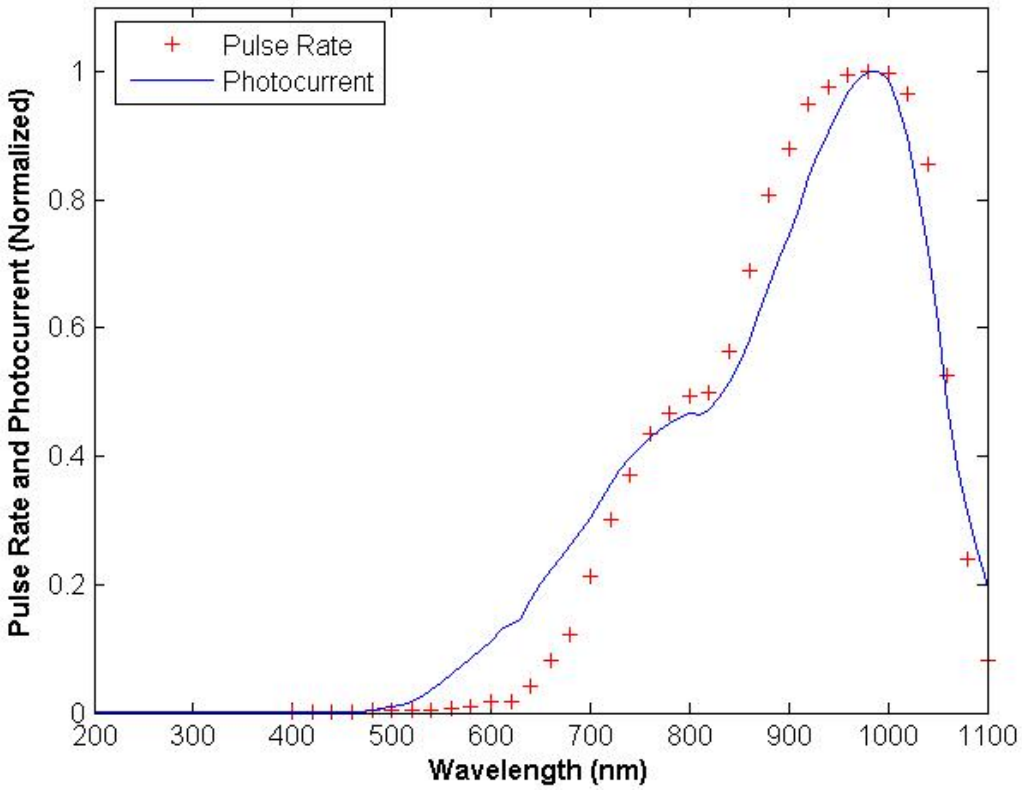


Figure 13. Comparison of the measured pulse rate and the photocurrent as a function of wavelength.

From the above similarity of the curves of the measured pulse rate and the photocurrent, one can conclude that it is feasible to construct pulse-mode detectors that are not limited to the Si photodiode spectral range, but can be extended to other wavelengths by replacing it with photodiodes made of other semiconductors. Also, since the SCR is made of Si, similar behavior could be observed by directly shining the light on to the middle p-n junction, allowing the circuit to function as a pulse mode optical sensor.

Figure 14 shows the measured pulse rate as a function of photocurrent. The pulse rate approaches the maximum at 1678 pulses/sec (red line) at saturation, where the inter-pulse interval approaches the refractory period, as it would be explained in the following section. The purple line with equation $y = 1209.37x - 472.71$ is the linear fit for the current range from 0.55 to 1.5 μA. The pulse rate is determined by the amount of current injected into the middle p-n junction via the gate terminal of the SCR, which is connected

to the middle p-layer. Recall from Chapter I, that the photocurrent adds to the hole current [5]. Figure 14 is another proof of this statement.

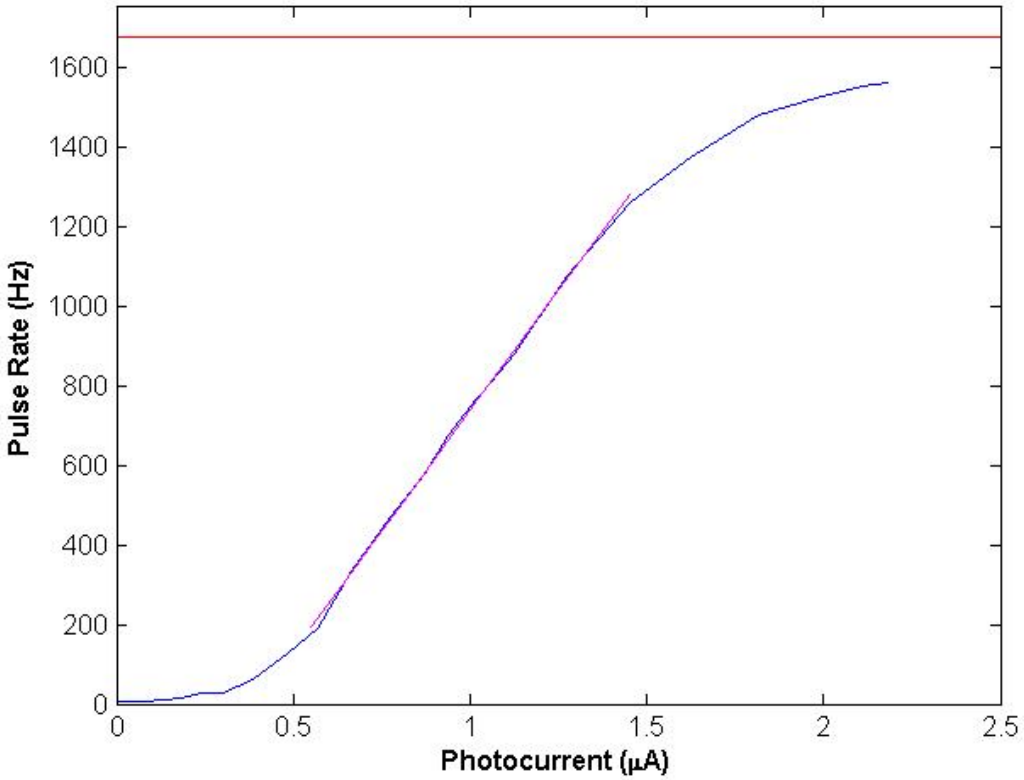


Figure 14. Measured pulse rate as a function of photocurrent (blue line). The red line corresponds to the maximum pulse rate (1678 pulses/sec).The purple line is the linear fit.

After the above measurements, the slit width was changed to 0.8 mm, keeping the bias the same at 16.1 V. This time, saturation occurred — the pulse rate was the same independent of the wavelength — at approximately 750 nm or 9 μW. During saturation all the statistical parameters (minimum and maximum value, mean and standard deviation) remained the same.

Figure 15 is a time interval distribution histogram for the set of pulses acquired from 400 to 1100 nm at 16.1 V. The number of pulses represented in the figure is approximately 21,000. This number corresponds to the total number of pulses measured. The histogram is normalized by dividing the number of counts in the bin with the total amount of counts for the specific interval, so that sum of the heights is unity. Fifty seven

percent of the pulses found to have pulse intervals below 1 msec. Recall that the RC constant of the circuit is 0.89 msec which sets the lower limit for the pulse intervals. In addition, pulse intervals up to 84 msec were observed (not shown in Figure 15), especially for shorter wavelengths where the incident power is relatively weak. However, probability of occurrence of such large time intervals is negligibly small.

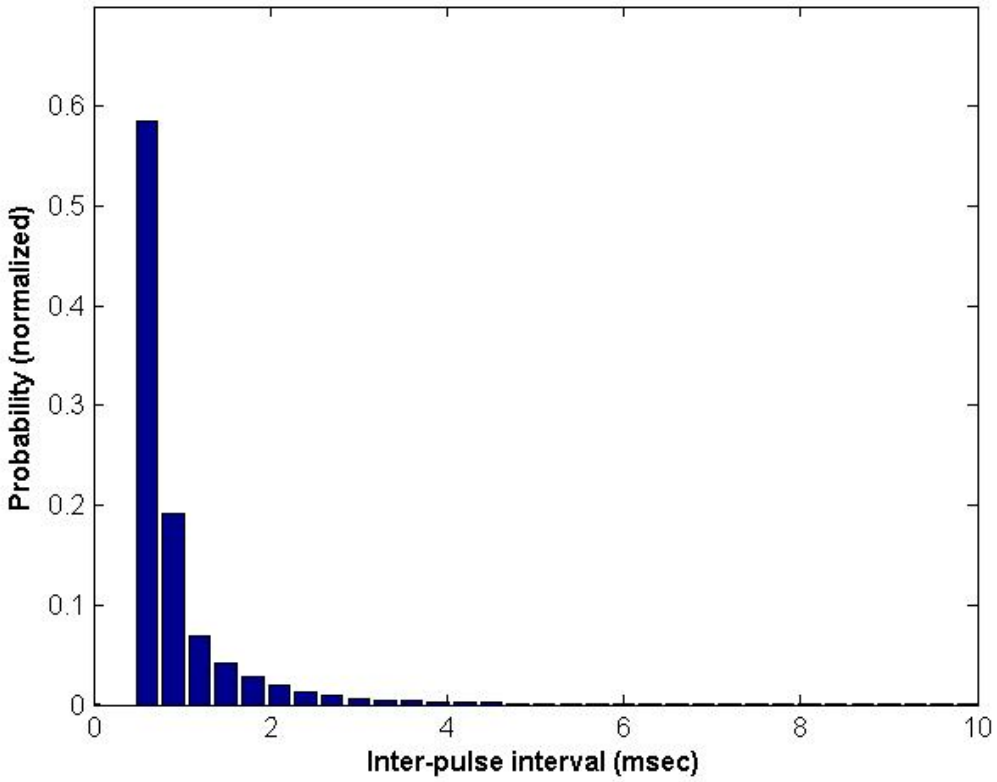


Figure 15. Histogram of inter-pulse interval distribution.

B. STATISTICAL MODEL

The objective of data collection was to measure the pulse rate dependence of a Si photodiode-SCR circuit on the intensity of the incident illumination. The best way to achieve this objective was to record the time intervals between pulses. The inter-pulse interval and also its mean and other necessary information for the statistical analysis that will follow were calculated by a MATLAB script, using the measured time intervals captured by a LABVIEW data acquisition program.

1. Distribution of Number of Pulses

The following analysis will show that the probability of pulse occurrence follows the Poisson distribution. The analysis will be based on the derivation of the general Poisson model found in [21].

The Poisson model is derived by considering the probability of pulsing between time t and $t+\Delta t$. Events (pulse occurrence) in the interval are considered to be independent from the previous history. In addition, as $\Delta t \to 0$, the probability of one event occurring in the time interval is proportional to the length of the time interval, i.e, the probability is $r\Delta t$, where r is the pulse rate. The probability that more than one event occurs in the small time interval is assumed to be zero.

For the time interval depicted in Figure 16, and using the fact that the events are independent:

$$\Pr[k \text{ events in } t+\Delta t]=\Pr[k-1 \text{ in } t]\times\Pr[1 \text{ in } \Delta t]+\Pr[k \text{ in } t]\times\Pr[0 \text{ in } \Delta t] \qquad (2.3)$$

or

$$\Pr[k \text{ events in } t+\Delta t]=\Pr[k-1 \text{ in } t]\times r\Delta t+\Pr[k \text{ in } t]\times(1-r\Delta t) \qquad (2.4)$$

Figure 16. Time interval for arrival of events (From [21]).

Rearranging, and taking the limit as $\Delta t \to 0$, yields:

$$\lim_{\Delta t\to 0}\frac{\Pr[k \text{ events in } t+\Delta t]-\Pr[k \text{ in } t]}{\Delta t}+r\Pr[k \text{ in } t]=r\Pr[k-1 \text{ in } t] \qquad (2.5)$$

or

$$\frac{d\Pr[k \text{ in } t]}{dt}+r\Pr[k \text{ in } t]=r\Pr[k-1 \text{ in } t] \qquad (2.6)$$

and the solution of this differential equation in given by

$$\Pr[k \text{ in } t]=\frac{(rt)^k e^{-rt}}{k!} \quad \text{where} \quad t \geq 0 \text{ and } k=0,1,2,... \tag{2.7}$$

Thus, the probability mass function of pulse occurrence follows the Poisson distribution, with k, the number of pulses in a fixed time interval t, to be a random variable. It is proven in [21] that the distribution of inter-pulse interval is the Erlang function, which is a special case of the gamma probability density function if k is an integer:

$$f_T(t) = \frac{r^k t^{k-1} e^{-rt}}{(k-1)!}, \quad t \geq 0 \tag{2.8}$$

which for $k = 1$ reduces to the exponential density function

$$f_T(t) = re^{-rt}, \quad t \geq 0 \tag{2.9}$$

and, therefore, the first order inter-pulse intervals of the Poisson model are exponentially distributed. The mean inter-pulse interval is given by

$$E\{T\} = \frac{1}{r} = < t > \tag{2.10}$$

hence the expected value of the inter-pulse interval equals the reciprocal of the pulse rate of the Poisson process.

Figure 17 depicts the average of the measured inter-pulse interval (red circles) in comparison with the reciprocal of the measured pulse rate (blue crosses). The data in Figure 17 confirm relationship based on the Poisson process $< t >= \frac{1}{r}$. The discrepancy observed, for wavelengths below 620 nm, may be due to the fact that only a small number of pulses were captured during the measurements as a result of low responsivity of the Si photodiode as well as weak intensity of the light source.

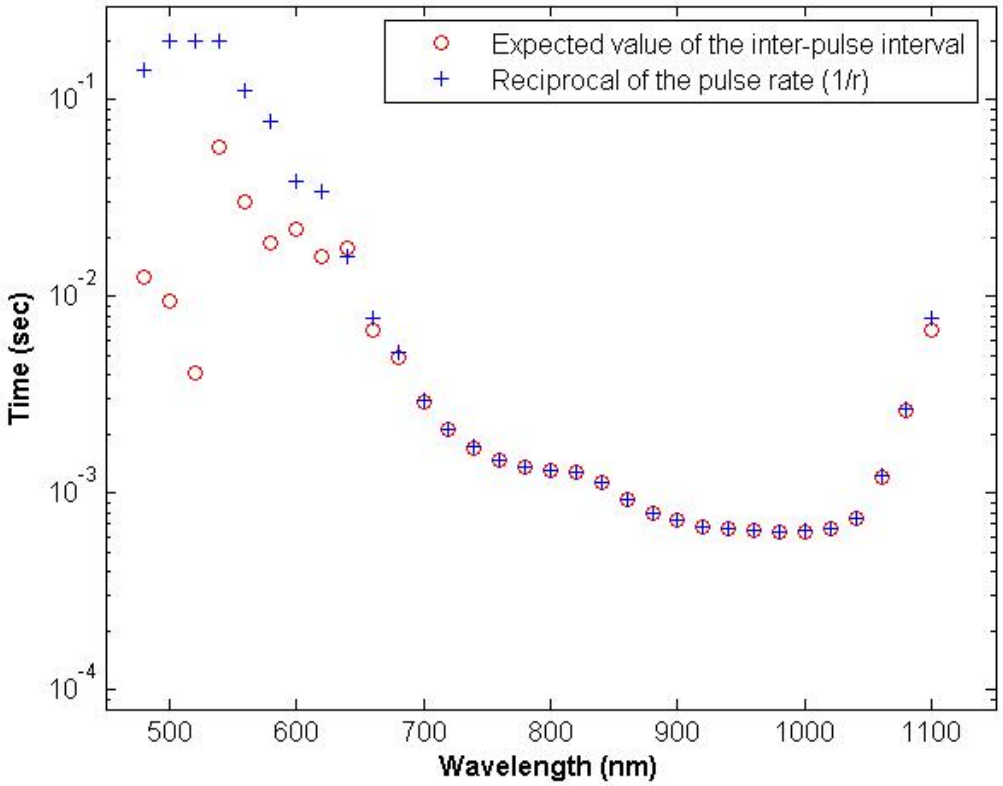


Figure 17. Comparison of the expected value of the measured inter-pulse interval with the reciprocal of the pulse rate (1/r) as a function of wavelength.

2. Refractory Period

Pulses were observed to have a width that was greater than zero, and specifically, it was determined that the inter-pulse interval was never less than 0.596 msec. This duration of the pulse is called the refractory period (see Figure 18), during which the probability of occurrence of the next pulse within this duration is zero. The refractory period is due to the RC time constant associated with the circuit. Based on the refractory period of 0.596 msec, it can be calculated that the maximum pulse rate is 1678 pulses per sec.

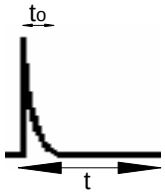


Figure 18. Inter-pulse time and refractory period t_o.

The refractory period is a behavior also observed in neurons [22]. A neuron cannot fire another pulse during the refractory period, until it has time to reset.

The effect of the pulse width can be taken into account by modifying equation (2.9) to read:

$$f_T(t) = \begin{cases} 0, & 0 < t < t_o \\ re^{-r(t-t_0)}, & t \geq t_o \end{cases} \tag{2.11}$$

Using equation (2.11), the probability density of having one pulse at t becomes

$$f_T(t) = \begin{cases} 0, & 0 < t < t_o \\ \dfrac{1}{<t>} e^{-\frac{t-t_0}{<t>}}, & t \geq t_o \end{cases} \tag{2.12}$$

Figure 19 is a plot of PDF fit to data for a series of wavelengths 700 nm (red), 800 nm (green), 900 nm (blue), and 1000 nm (purple). The solid lines correspond to the calculated probability density using the measured pulse rate for each wavelength and the "exppdf" function in MATLAB [23]. The crosses correspond to the calculated inter-pulse interval using the PDF of equation (2.12).

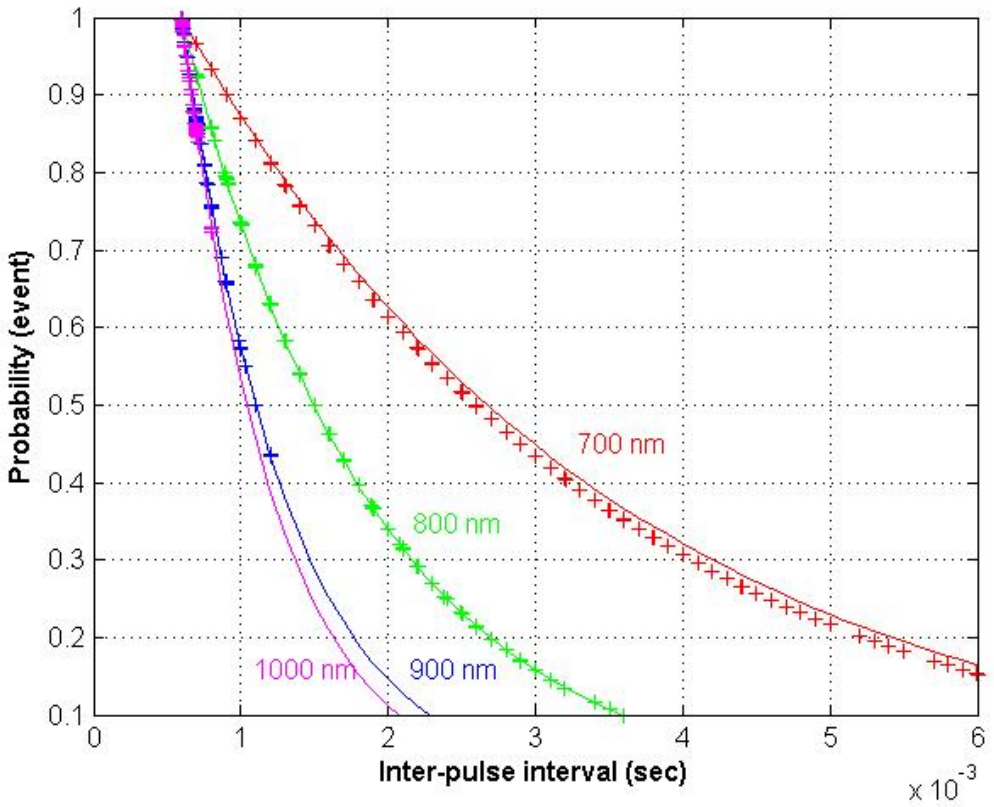


Figure 19. Comparison of the PDF of equation (2.12) (crosses) with the theoretical exponential PDF (solid lines) for a series of wavelengths.

The mean inter-pulse interval reduced as the wavelength increased from 400 to 980 nm (maximum pulse rate) and then increased as the wavelength approached the $\lambda_{cut\,off}$ of Si due to the variation of responsivity of the photodiode. The inter-pulse interval for 1000 nm (near the maximum of the pulse rate at 980 nm) approaches the refractory period. At lower wavelengths (lower responsivity), the rate of injection of holes in the middle p-n region is relatively small, which is responsible for longer and aperiodic inter-pulse intervals.

The PDF fit is found to agree well for all of the wavelengths. This suggests that the pulse generation, in the first order statistics, is a memoryless process described by the Poisson distribution with a refractory period.

3. Higher Order Statistics and Renewal Process

In the previous section, only the first order statistics were mentioned. In order to account for higher order statistics (many pulses in time t), the Poisson process should be replaced by a renewal process [24]. In this case, the Poisson distribution cannot account for the refractory period for two or more pulses, because it would give a finite probability of occurrence for the refractory period for $k > 1$, which of course is not true. In addition, a Poisson process could not account for the regularity of the pulses seen during saturation. When the circuit pulses near its maximum rate, this refractoriness causes the pulse train to become more regular than a Poisson process with the same rate [22]. Thus, the firing probability depends not only on the stimulus, but also on the preceding pulse train.

A stochastic process $\{N(t),\, t \geq 0\}$ is called a counting process if $N(t)$ represents the total number of events that have occurred up to time t. It is called a renewal process if the inter-arrival times $T_1, T_2, \ldots$ are independent identically distributed (IID) non-negative random variables. The instants $t_k = \sum_{j=1}^{k} T_j$, where $k = 1, 2, \ldots$ are called renewal times. The renewal process is defined by the counting process as [25]

$$N(t) = \begin{cases} 0 & \text{for } 0 \leq t < t_1 \\ \max\{k \mid t_k \leq t,\, k = 1, 2, \ldots\} & \text{for } t \geq t_1 \end{cases} \tag{2.13}$$

and $N(t)$ is thus the random number of renewals occurring in (0,t].

The most commonly known renewal process is the Poisson process, and thus the Poisson process can be replaced by a renewal process to take into account for the refractory period [24]. In this case, the inter-pulse interval is given by the gamma probability density function of equation (2.8). When $k = 1$, $f_T(t)$ becomes the exponential distribution (see equation (2.9)). When k is infinity, $f_T(t)$ is a distribution with zero variance and the pulse train is perfectly regular. This behavior was observed during saturation. The mean inter-pulse interval is again $< t >= \frac{1}{r}$. Studies have shown

that the inter-pulse interval of pulse trains recorded from retinal ganglion cells can be very well modeled by renewal processes with gamma-distributed intervals [22, 24].

Figure 20 depicts the histogram for pulse intervals observed at 800 nm superimposed with the theoretical gamma PDF (red line) using the "gampdf" function in MATLAB [26]. The histogram is normalized by dividing the number of counts in the bin with the total amount of counts for the specific interval, so that sum of the heights is unity.

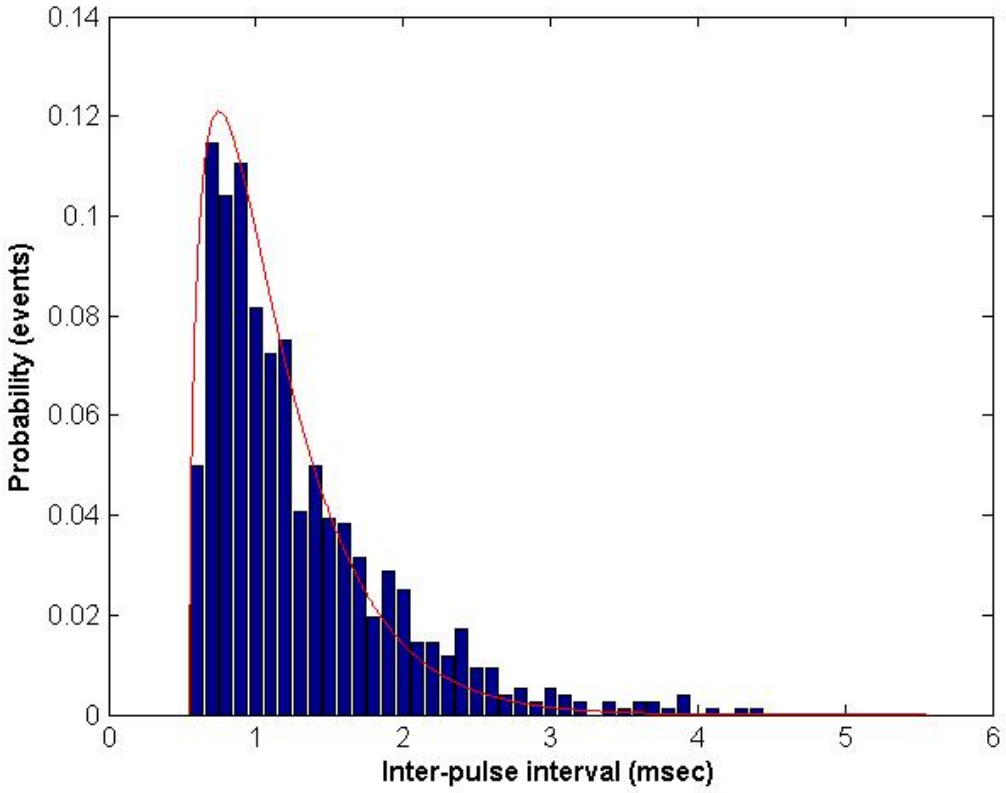


Figure 20. Histogram of the inter-pulse interval for 800 nm compared with the theoretical Gamma Probability density function (red line).

From Figure 20, it can be stated that the inter-pulse interval follows the inter-pulse interval of a renewal process. The same shape was also seen at other wavelengths. Thus, for higher order statistics, a renewal process should replace the Poisson process.

C. SUMMARY

This chapter discussed the observed pulse rate dependence on the intensity of the incident monochromatic illumination on a FDS 100 photodiode that was connected to an SCR circuit. It was determined that at low power (0 - 6 μW) of incident illumination, the pulse rate follows the photocurrent response of the photodiode, concluding that by directly shining the light on to the middle p-n junction, the circuit will function as a pulse mode light sensor. In addition, it is also feasible to construct pulse mode sensors for other wavelengths. In contrast, at higher powers (over 9 μW), saturation occurs and the pulse rate stays the same, independent of the wavelength of the incident light.

In addition, in first order statistics, the behavior observed can be characterized using a Poisson process. For higher order statistics, though, a renewal process should replace the Poisson process. Since the first order statistics are also included in the renewal process, a renewal process statistical model can be used to analyze the circuit's behavior.

Finally, the produced pulses were observed to have similar characteristics (periodicity, refractory period, and burstiness) as the spikes produced by the retinal neurons to the presence of external stimuli.

The next chapter will present the basic concepts of neural networks and the description of the widely used back-propagation algorithm.

THIS PAGE INTENTIONALLY LEFT BLANK

III. NEURAL NETWORKS

This chapter presents the basic concepts of neural networks: the neuron model, the neural networks architecture, and the learning process, a process that is of primary significance for a neural network. It is followed by the description of the commonly used back-propagation algorithm.

A. INTRODUCTION

The brain is a highly complex, nonlinear, and parallel computer. It has the capability to organize its structural constituents, known as neurons, so as to perform certain computations, (e.g., pattern recognition and motion control), many times faster than any computer today [27].

Computers cannot solve all the real-world applications, due to their sequential processing. Thus, features from physiology of the brain are used as the basis of the processing models, called neural networks [28].

A neural network is made up of simple processing units (neurons), and models the way in which the brain performs a particular task of interest. A neural network derives its computing power through its parallel structure, and its ability to learn and therefore generalize — produce reasonable outputs for inputs not encountered during training [27]. Although there are many similarities to the human brain, we cannot think of neural networks as a way to mimic the human brain.

1. Human Brain and Neurons

The human cortex has approximately 10 billion neurons. Neurons come in a wide variety of shapes and sizes [27]. Each neuron has a potential difference of 70 to 100 mV across its membrane, which is called the resisting potential. Connections to the neuron, known as synapses, from other neurons occur at various locations on the neuron's cell [28], which convert the pro-synaptic electric signals to chemical and then to post-electric signals. It is estimated that there are over 60 trillion synapses in the human cortex [27].

Nerve impulses can result in local changes in the potential of the receiving neuron. The input potentials are summed and if the depolarization is sufficient, an action potential or spike is generated. Once an action potential is created, the neuron is incapable of firing another spike for about 1 msec, until it restores its resisting potential (refractory period) [28].

2. Neuron Model

A neuron is the processing unit of a neural network. Like a real neuron, it has many inputs, but only one single output, which may be the input to other neurons in the network [28].

Figure 21 shows the model of a neuron [27]. The neuron k is actually a summing junction. It sums the product of an input x_j with its corresponding synaptic weight (strength) w_{kj} and an externally applied bias b_k. The bias can increase or lower the output of the neuron, depending on whether it is positive or negative, respectively.

The output of the summing junction υ_k is the input to an activation function φ, which limits the output of a neuron to some finite value y_k. Typically, y_k belongs in the closed interval [0, 1] or alternatively [-1, 1].

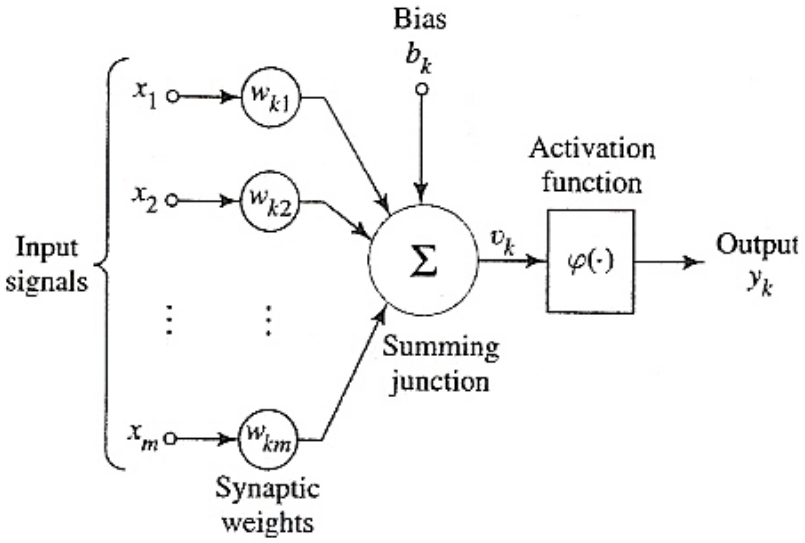


Figure 21. The Non-linear Neuron model (From [27]).

In mathematical terms, we may describe a neuron k with the following equations [27]:

$$\upsilon_k = \sum_{j=1}^{m} w_{kj} x_j + b_k \tag{3.1}$$

and

$$y_k = \varphi(\upsilon_k) \tag{3.2}$$

There are many types of activation functions, but the most common is the sigmoid function [27]. An example is the logistic function (see Figure 22), defined by

$$\varphi(\upsilon) = \frac{1}{1 + e^{-\alpha\upsilon}} \tag{3.3}$$

where α is the slope parameter of the sigmoid function. A sigmoid function limits the output of the neuron to the interval [0, 1]. In this thesis, the sigmoid function for $\alpha = 1$ will be used.

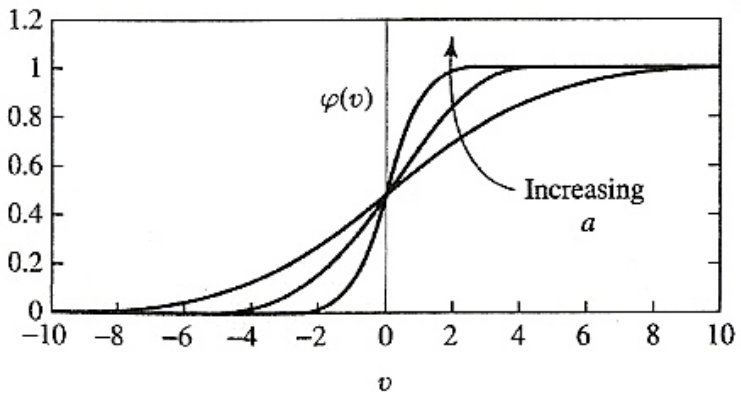


Figure 22. Sigmoid function for varying slope parameter α. (From [27]).

3. Network Architectures

In a neural network the neurons are organized in the form of layers. There is an input layer that projects onto one or more hidden layers or directly onto an output layer. Hidden neurons are considered to "extract" the features of the input patterns (vectors), and thus characterize the training data [27]. Such a network is called feed-forward.

The input layer consists of the elements of the input vector. No computation is performed at the input layer. Then, the outputs of the input layer are the inputs to the neurons in the second layer (i.e., the first hidden layer). The output signals of the second layer are used as inputs to the third layer, and so on for the rest of the network. The output of the neurons in the output layer of the network is the overall response of the network. The graph in Figure 23 illustrates the layout of a multilayer feed-forward neural network for the case of a single hidden layer. It is said to be fully connected when every node in each layer of the network is connected to every other node in the adjacent forward layer [27].

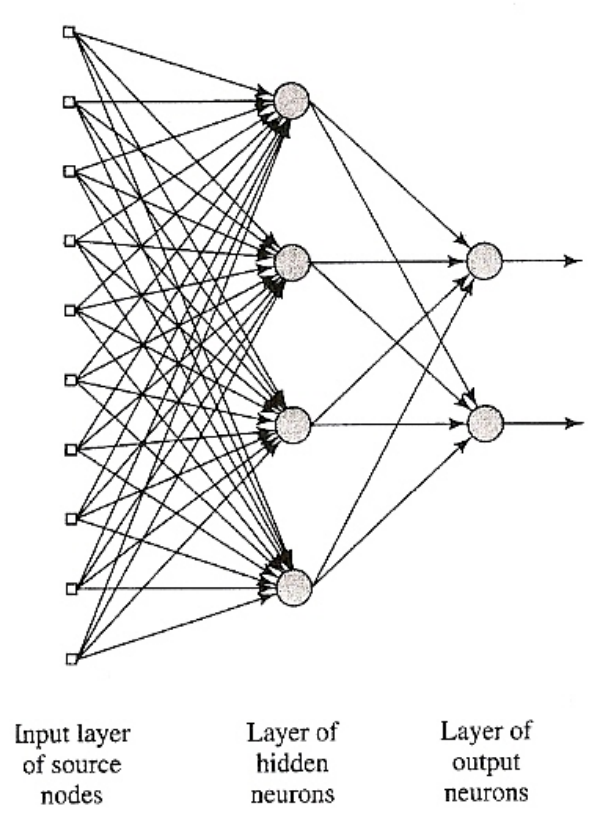


Figure 23. Fully connected feed-forward network with one hidden and one output layer. (From [27]).

4. Learning Process – Supervised Learning

The property that is of primary significance for a neural network is its ability to learn from its environment and to improve its performance through learning, using a prescribed measure. A neural network learns about its environment by adjusting its synaptic weights and bias levels [27].

An algorithm used to solve a specific problem is called a learning algorithm. Of course, there is no unique learning algorithm for the design of neural networks. Rather, each learning algorithm is application oriented, and especially, as described in the following section, influenced by the learning task.

One of the most common learning processes is learning with a teacher, or supervised learning [27]. The teacher has knowledge of the environment, in the form of input-output vectors, but the neural network does not. When the teacher and the neural

network are both exposed to the same training vector from the environment, the teacher provides the desired response to the neural network, which then adjusts its parameters (weights and biases) in order to match the desired response. After many iterations of training vectors from the environment the neural network is trained and all its parameters are then fixed. When this happens, the neural network can deal with the environment completely by itself.

5. Learning Tasks – Pattern Recognition

The choice of the particular learning algorithm is influenced by the learning task that a neural network will perform. Some learning tasks include pattern association (associative memory), pattern recognition, function approximation, control, filtering, beamforming [27] and many others. Pattern recognition is the main focus of this thesis and it will be described further.

Humans are good at pattern recognition. We receive data from the enviroment around us via our senses and are able to recognize the source of the data, almost immediately. Humans perform pattern recognition through a learning process; so it is with neural networks [27].

Pattern recognition is defined as the process whereby a received pattern/signal is assigned to one of a prescribed number of classes (categories) [27]. In the following chapter, a detector will be trained, using a neural network model, to indentify the incident pattern, which will be one of the numerical characters 0 through 9.

B. BACK PROPAGATION

One of the well known algorithms to solve complex problems by training the network in a supervised manner is the error back-propagation algorithm. Back propagation neural networks have been used to solve various problems, like pattern recognition, paint-quality inspection of automobile panels, and even classification of sonar signals [28].

The error back-propagation learning consists of two passes in the network: a forward pass and a backward pass. In the forward pass, a pattern (input vector) is applied to the input layer of the network and the actual response of the network is produced at the output layer. During the forward pass the synaptic weights of the network are all fixed. During the backward pass, on the other hand, the synaptic weights are adjusted in accordance with an error-correction rule. Specifically, the actual response of the network is subtracted from a desired (target) response to produce an error signal. This error signal is then propagated backward through the network— hence the name "error back-propagation". The synaptic weights are adjusted to make the actual response of the network become closer to the desired response [27]. Figure 24 illustrates the direction of the two signal flows.

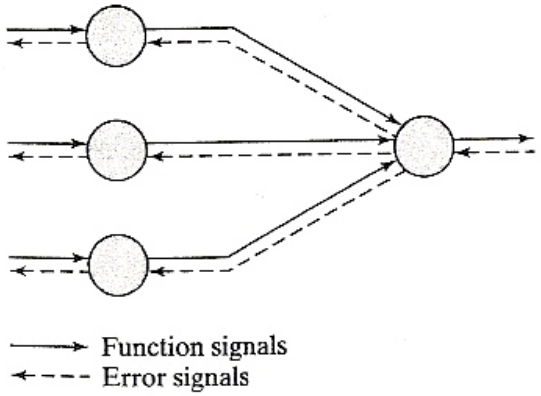


Figure 24. Illustration of the directions of the two signal flows: forward propagation of function signals and back-propagation of error signals. (From [27]).

Back propagation algorithm is based on the error-correction learning rule. The error signal at the output of neuron j of the p^{th} training example is defined by [27]:

$$e_j(p) = d_j(p) - y_j(p) \tag{3.4}$$

where $e_j(p)$ is the error signal, $d_j(p)$ is the desired response, and $y_j(p)$ is the output of the j^{th} neuron at iteration p.

The average squared error energy E_{avg} is the measure of the learning performance. The goal of the learning process is to adjust the free parameters of the network to minimize E_{avg}. It is defined as [27]

$$E_{avg} = \frac{1}{N}\sum_{p=1}^{N}\sum_{j=C} e_j^2(p) \tag{3.5}$$

where C is the number of neurons in the output layer and N is the total number of patterns of the training set.

However, each neuron in a hidden layer receives only a portion of the total error signal, based roughly on the relative contribution the neuron made to the original output [28]. The mathematical formulas for calculating how the weights and biases are adjusted is beyond the scope of this thesis and it will not be mentioned.

C. SUMMARY

This chapter presented the basic concepts of neural networks. In the next chapter, a specific problem of pattern recognition will be addressed and solved using a neural network.

IV. APPLICATION OF THE PULSING CIRCUIT AND NEURAL NETWORKS

In this chapter the unique characteristics of the pulse generating circuit will be used to construct a sensor array capable of recognizing visual patterns (numbers from 0 to 9) using a neural network, and thus modeling the human visual cortex.

A. EXPERIMENTAL SET-UP

The functional block diagram of the experimental setup used in this work is shown in Figure 25.

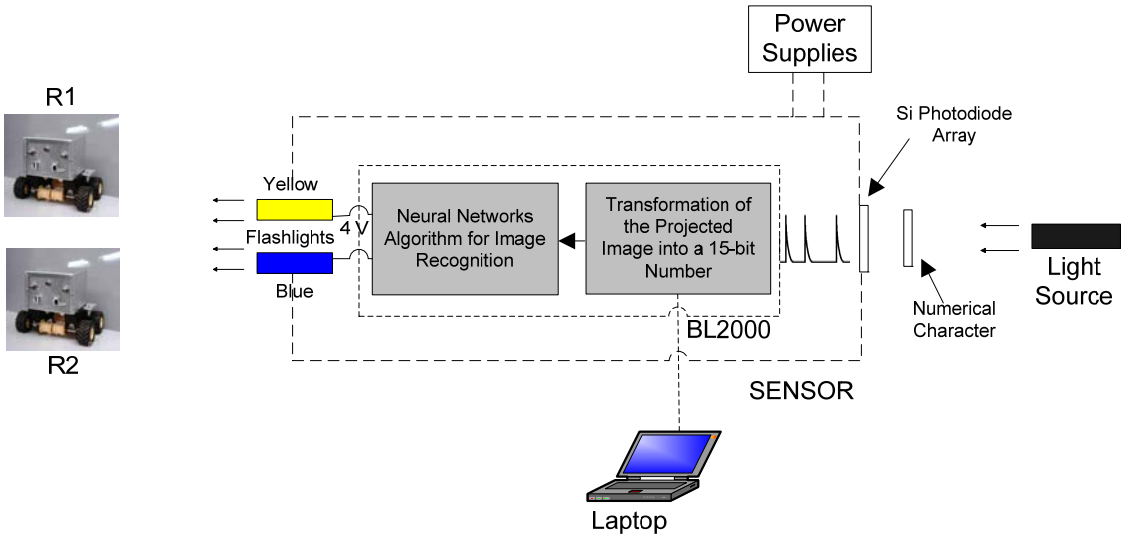


Figure 25. Functional Block Diagram of the experimental setup.

The system consists of a variable-intensity light source which produced the necessary light for the projection of the object being recognized (a numerical character) onto the 5×3 Si photodiode array. Each photodiode was connected to a SCR circuit. The ten numerical characters, 0 to 9, were cut into hard paper to let the light pass through. Two of the numerical characters used for this purpose, numbers 5 and 7, as well as a noisy number 4, can be seen in Figure 26.

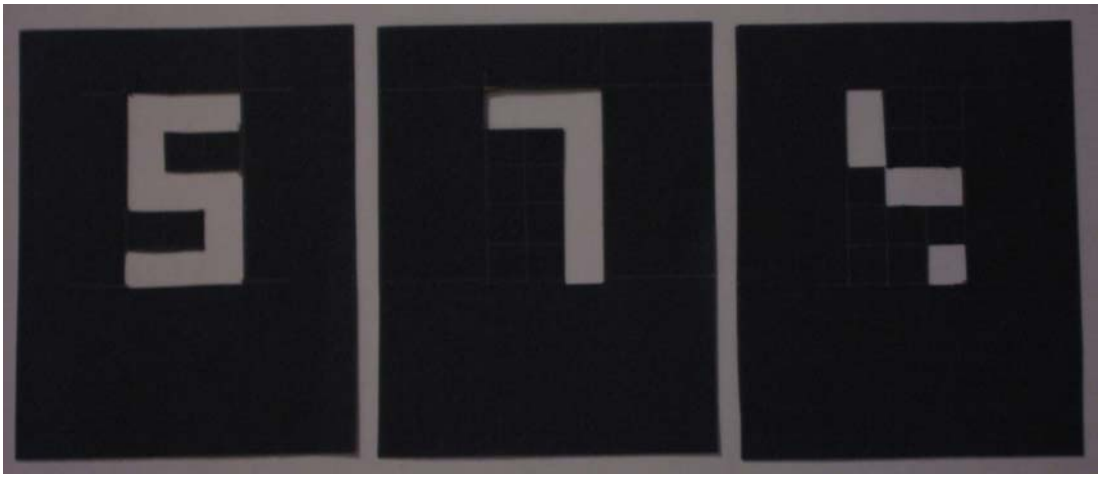


Figure 26. Some of the numerical characters used in the experiment.

When light was incident on a photodiode, the corresponding circuit produced a pulse train. In contrast, when the light was blocked by the paper the corresponding photodiode did not produce enough photocurrent to force its circuit to pulse. For example, the illumination of number seven is depicted in Figure 27.

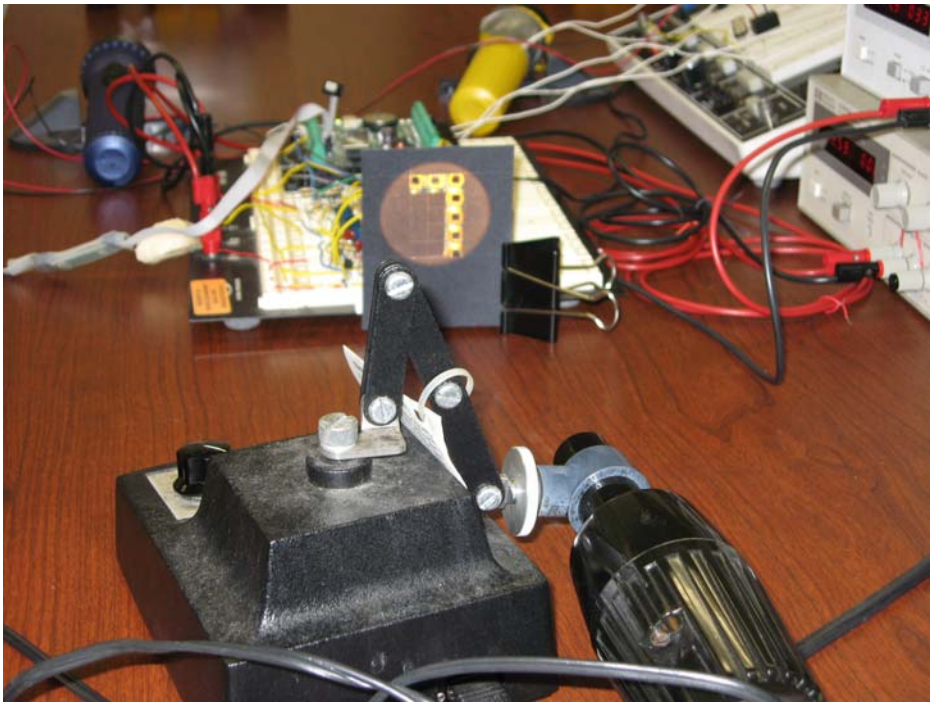


Figure 27. Illumination of number seven on the photodiode array.

A Wildcat BL2000 microprocessor was used to measure the output voltage of each circuit and transformed it to a 15-bit binary number. This 15-bit number was the input to the neural network algorithm which then recognized the number being illuminated at the photodiode array. The last task of the BL2000 microprocessor was to output 4 V, depending on the recognized number, on one or both its outputs. This voltage drove the circuitry that controlled (switched ON) the corresponding flashlight. The light produced from the flashlight was incident on the corresponding robot's sensors (two photodiode-SCR circuits) forcing it to approach the sensor, and thus emulating a form of networked communication. The mapping for this operation can be seen in Table 2.

RECOGNIZED NUMBER	**FLASHLIGHT ON**	**ROBOT APPROACHING**
1, 2, 3	Yellow	R1
4, 5, 6	Blue	R2
7, 8, 9	Yellow & Blue	R1 & R2
0 or not valid number	-	-

Table 2. The mapping according to the recognized number.

In addition, two power supplies provided the necessary bias for the operation of the sensor. The first one provided 12.6 V to the SCR circuits, and the second one provided 15 V to the voltage followers used for interfacing the sensors with the BL2000. Finally, a computer was connected to the BL2000 via a RS232 link. The computer sent commands to BL2000 ("Compile" and "Run"), and monitored the correct recognition of the illuminated numerical character. The Wildcat BL2000 microprocessor could also operate in a stand-alone mode, but this way the recognized numbers could not been monitored.

A photograph of the complete experimental set-up is shown in Figure 28. More details about the various components will be provided in the following sections.

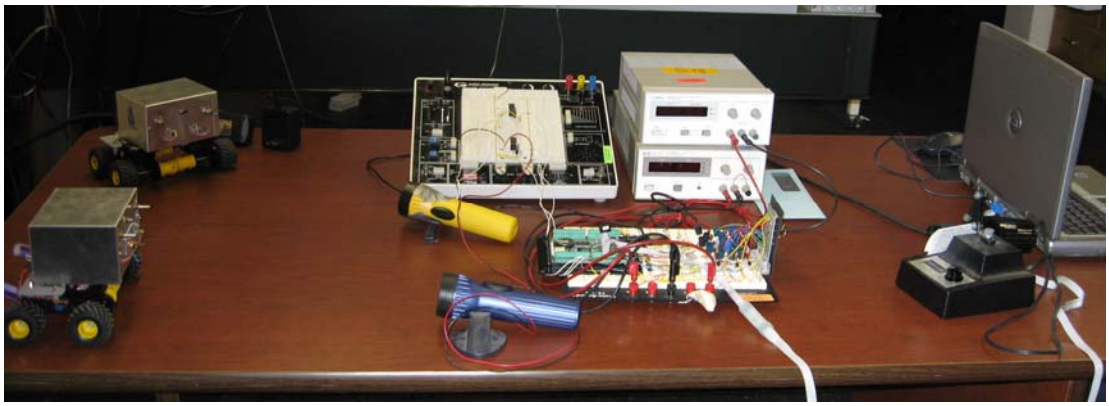


Figure 28. The set up for the application.

B. CONSTRUCTION OF THE SENSOR ARRAY

The sensor array consisted of 15 pulse-mode SCR-photodiode detectors, the Wildcat BL2000 microprocessor and two flashlights. The pulse-mode sensor array, the BL2000, as well as the supporting circuitry for interfacing them, were placed on a common circuit board (see Figure 29).

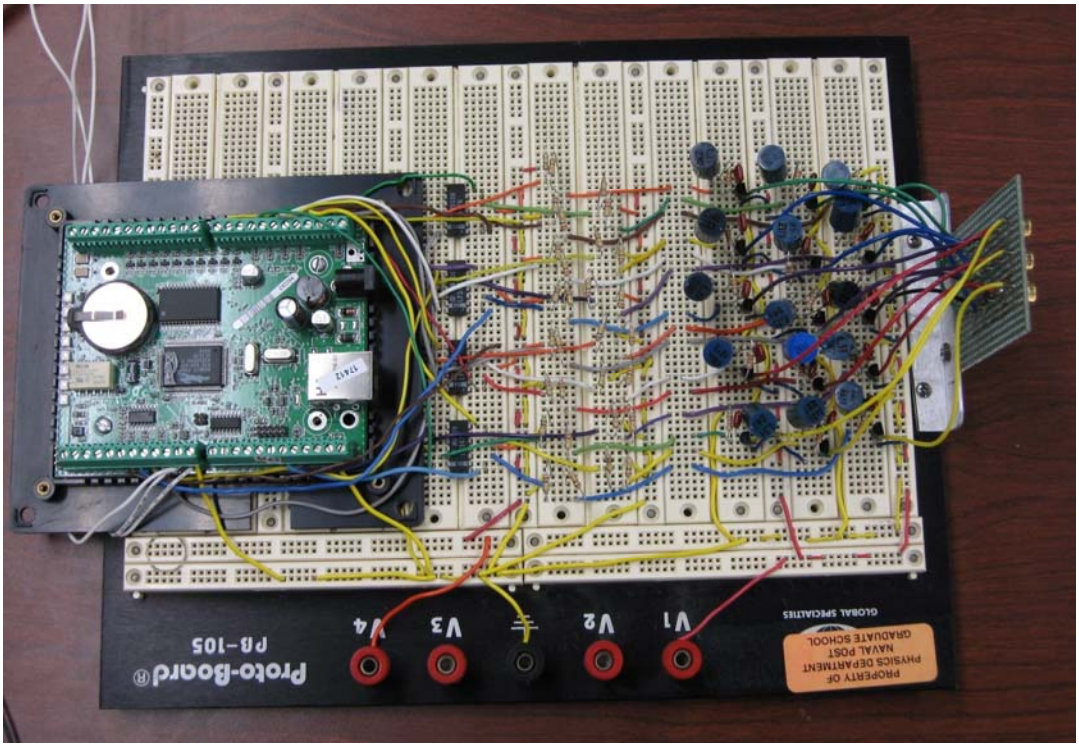


Figure 29. The sensor array with microprocessor.

1. The Sensor Pixel

A sketch of the circuit of a sensor pixel with interfacing electronics to the microprocessor is shown in Figure 30. Fifteen of these circuits were constructed.

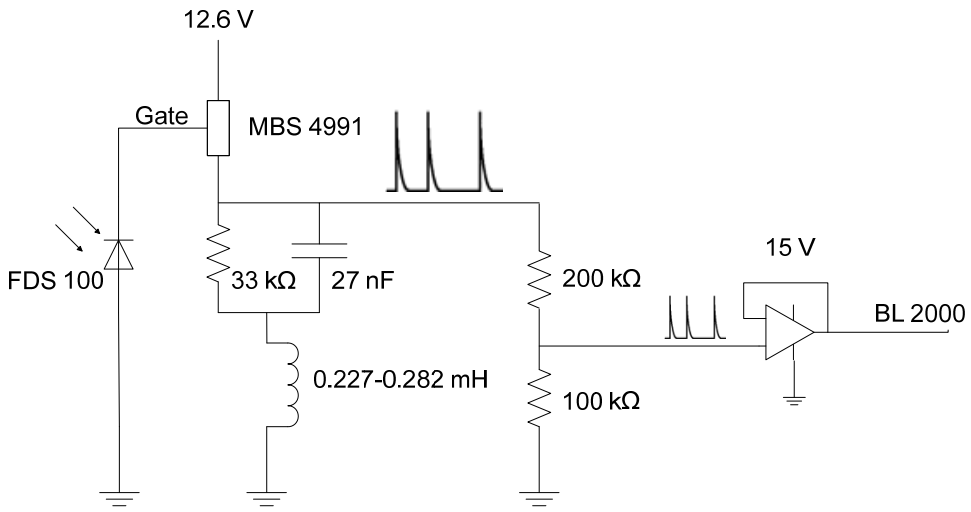


Figure 30. Sensor Circuit Schematic.

The necessary circuitry consisted of the following:

a. Fifteen FDS 100 Si photodiodes were arranged in a 5×3 planar array. The anodes of the photodiodes were connected together to provide a common ground. A cable was attached to each cathode, so that it could be connected to the corresponding pulse-mode circuit. The photodiode array was mounted perpendicular to the circuit board as depicted in Figure 31.

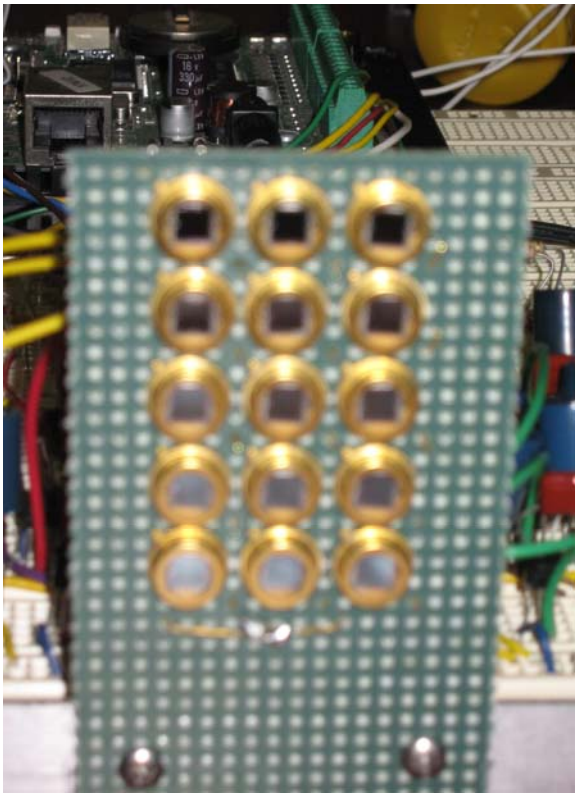


Figure 31. The Si photodiode array.

The retina's task is to convert an optical image into a neural image for transmission down the optic nerve for further analysis [27]. The processing in the primary visual cortex consists mainly of an orientation response; that is a response determined by the directions of lines and edges in the visual image [29]. This is how humans perceive the shape of an object. When an object enters the visual field, the neurons sensitive to the orientation of the edges of the object will respond by firing a number of spikes. The photodiodes of the sensor model mimic this form of behavior.

b. Fifteen circuits containing SCR and parallel RC combination were constructed, one for each photodiode. The cathode from each photodiode was connected to the gate of the corresponding SCR. The SCRs used were MBS 4991. The resistors were 33 kΩ and the capacitors used were 27 nF. The SCRs were selected to pulse independently within the range of 16.1 to 16.9 V bias. When the circuits were connected to the photodiode array and the common bias, it was found that all the circuits were pulsing together independent of whether light was incident on the corresponding photodiode or even if the photodiodes were not connected. This behavior was due to high

current surge during the pulsing which propagated through the ground to the other circuits, causing all the circuits to pulse together; this behavior was observed even when one circuit was pulsing. The above effect was minimized by putting 0.227 to 0.282 mH inductors on every circuit in series with the parallel RC and the ground as shown in Figure 30. The high impedance of the inductors to the surge prevented the synchronous pulsing of all the circuits.

c. Voltage dividers. The maximum voltage that was produced by the SCR circuits when they pulsed was approximately 17 V. For this voltage only 5 of the pins of the Wildcat BL 2000 could be used. In order to facilitate the 15 inputs simultaneously, 15 voltage divider circuits were made to reduce the maximum voltage below 10 V. Two resistors, 200 kΩ and 100 kΩ (see Figure 30) were used to reduce the voltage to 1/3 of its original value.

d. Voltage followers (buffers). The Wildcat BL 2000 had low input impedance that loaded down the circuits. In order to avoid this result, fifteen voltage followers (buffers) were made. The voltage follower has a gain of 1, very high input impedance, and at the same time very low output impedance [30].

2. The Wildcat BL 2000 Microprocessor

A Wildcat BL2000 microprocessor was used to measure the output voltage of each circuit and transformed it to a 15-bit binary number; 1 if it was above a voltage threshold and 0 if the voltage was below the threshold. The default value for the threshold is 2.4 V [31]. Thus, the first function of the BL2000 was to transform the image of the numerical character into a 15-bit number.

The 15-bit number was then input to the neural network algorithm that run in the BL2000 using Dynamic C. The result from this program was the recognition of the number created at the photodiode array. The details of the algorithm will be described in a following section.

The BL2000 also output 4 V depending on the recognized number on one or both its outputs (DAC0 or DAC1 or both). This voltage drove the circuitry that switched ON

the two flashlights for the corresponding robots. If the illuminated number was one, two or three, the BL2000 output 4 V to the yellow flashlight (see Table 2). If the number was four, five or six, 4V were applied to the blue flashlight. If the number was seven, eight or nine, a 4 V output was applied to both flashlights forcing both robots to approach the sensor. Finally, if the incident pattern corresponded to number zero or if the incident number did not correspond to a valid number, no voltage would be output to the flashlights, and thus none of the robots would move.

3. The Flashlight Circuitry

The flashlight circuitry can be depicted in Figure 32. Two identical circuits were made, one for each flashlight.

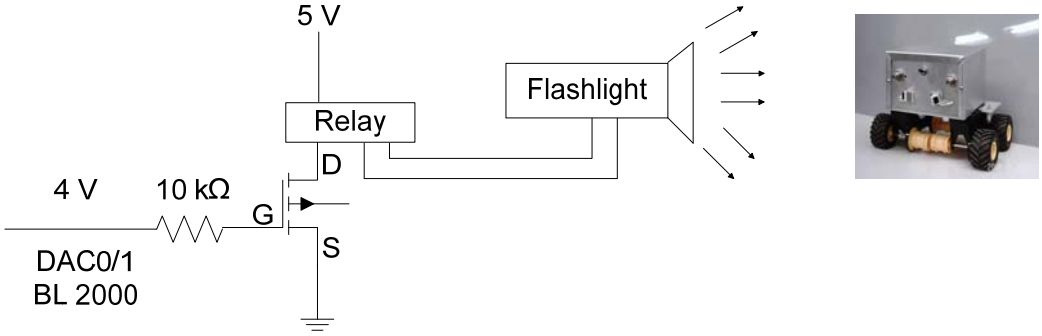


Figure 32. Flashlight Circuit Schematic.

When a small voltage is applied at a MOSFET's gate lead, the current flow through its drain-source channel is altered [30]. The output voltage from the BL2000 drove the MOSFET that produced the necessary current to switch on the relay and thus the flashlight. Light from the flashlight was incident on the corresponding robot forcing it to move and approach the sensor.

C. THE ROBOTS

The two robots were constructed by Matos [4]. The robot platform can be seen in Figure 33. Two FDS 100 Si photodiodes play the role of the robot's sensor. Using the unique characteristics of the pulse generating circuit, each robot can track, turn and

follow the light source that is incident on the photodiodes. Using analog electronics, the produced pulses are converted to rectangle pulses that drive the two DC motors which control each side of the wheels. More detail on the construction of the robots can be found in [4].

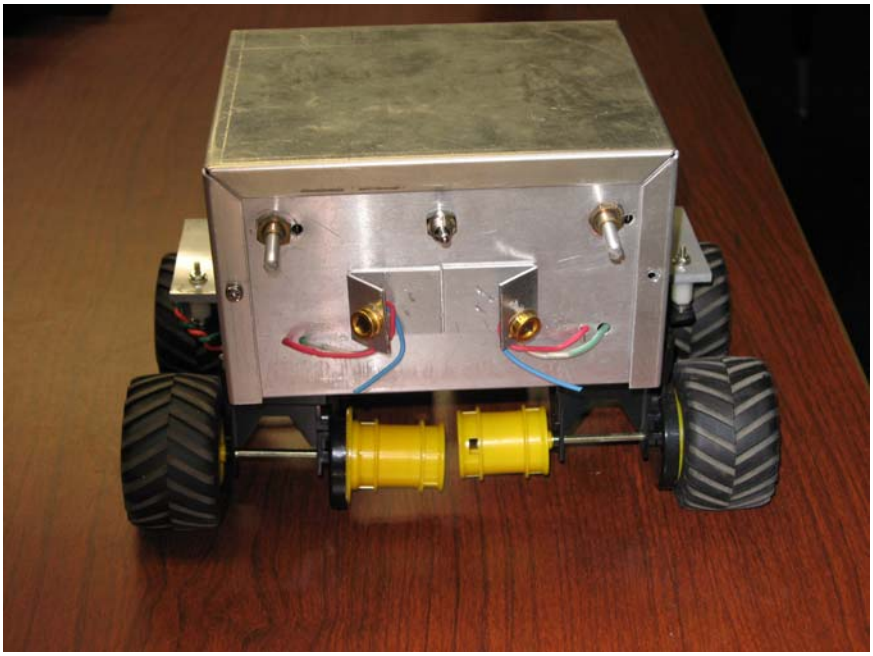
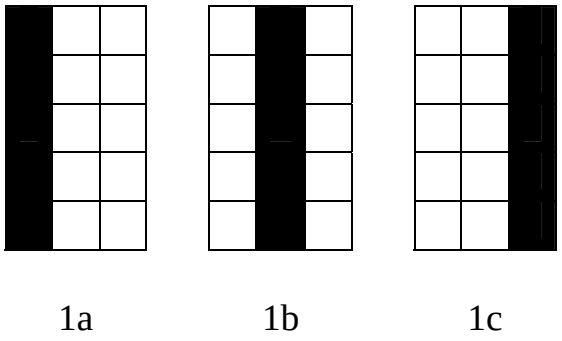


Figure 33. Robot platform.

D. REPRESENTATION OF THE NUMERICAL CHARACTERS

It was found that the numbers should be in the form of a 15-bit binary number in order to be easily interpreted by the program that determined what the incident number was. The ten numbers and their corresponding 15-bit binary representations are depicted in Figure 34. Except for number one that has three possible choices, each number is centered in the system's field of vision and occupies the whole area of the photodiode array. The 15-bit binary numbers were generated by moving the three columns of the array to a single row and setting 1 for light and 0 for dark.

A 12×15 matrix consisting of the above twelve 15-bit binary numbers formed the library of a training vector generation algorithm described in the following section.

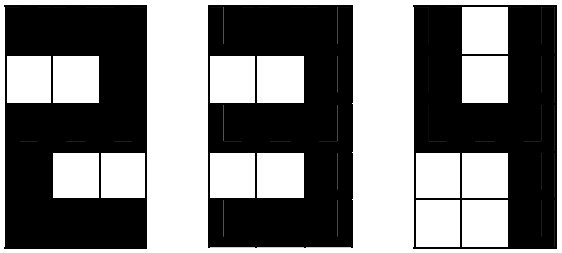


1a 1b 1c

1a : 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0

1b : 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0

1c : 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1

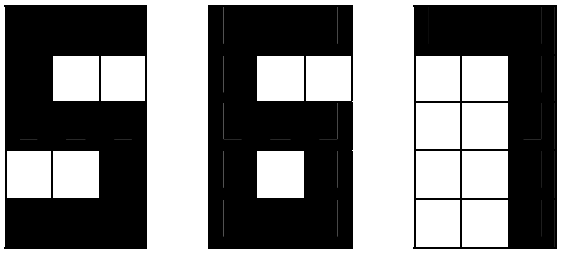


2 3 4

2 : 1 0 1 1 1 1 0 1 0 1 1 1 1 0 1

3 : 1 0 1 0 1 1 0 1 0 1 1 1 1 1 1

4 : 1 1 1 0 0 0 0 1 0 0 1 1 1 1 1

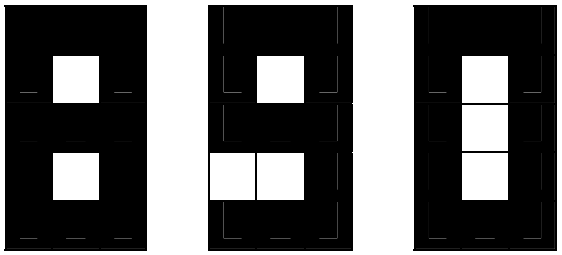


5 6 7

5 : 1 1 1 0 1 1 0 1 0 1 1 0 1 1 1

6 : 1 1 1 1 1 1 0 1 0 1 1 0 1 1 1

7 : 1 0 0 0 0 1 0 0 0 0 1 1 1 1 1



8 9 0

8 : 1 1 1 1 1 1 0 1 0 1 1 1 1 1 1

9 : 1 1 1 0 1 1 0 1 0 1 1 1 1 1 1

0 : 1 1 1 1 1 1 0 0 0 1 1 1 1 1 1

Figure 34. The ten numbers and their 15-bit binary representations. Black pixels represent light (1) while white pixels represent dark (0).

E. THE NUMBER RECOGNITION ALGORITHM

The BL2000 transformed the image of the numerical character from 0 to 9 into a 15-bit number. An algorithm should determine what that numerical character was. Although a lookup table approach would be reasonably fast, there are many situations that could occur in this particular system that could not be handled by that method. Consider, for example, the two kinds of possible errors that could happen in this system. First, a photodiode that should pulse, did not, creating a missing pixel in the representation of the number in Figure 34. The second case could be a circuit with photodiode in the dark creating a pulse. A single pixel error would cause the lookup algorithm to return either a null or the wrong number, since the match between the input and the target pattern should be exact. To account for these real world scenarios an algorithm that is flexible should be implemented.

1. Training Vector Generation

The neural network that solved this pattern recognition problem was trained using the MATLAB Neural Network Toolbox [32]. The required input-target (training) vectors were produced by an algorithm, named training vector generation algorithm, for distinguishing it from the neural networks algorithm. The training vector generation algorithm was used to automatically produce the input and target vectors for the training of the neural network. The code for the training vector generation algorithm, written in MATLAB, is included in Appendix A and the flow chart of it can be seen in Figure 35.

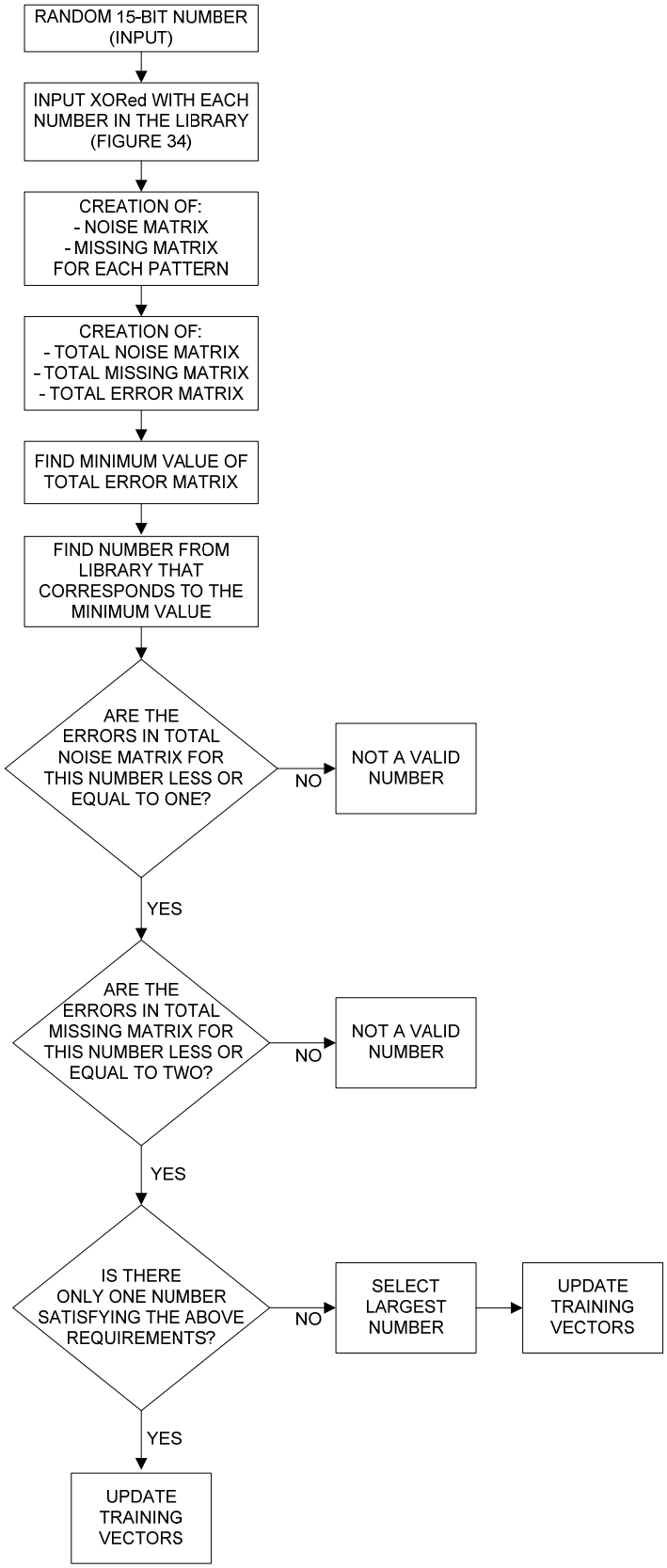


Figure 35. Flow chart of the training vector generation algorithm.

The training vector generation algorithm is described below. Using a “for” loop, a set of random 15-bit numbers (inputs) were created. Each input was XORed with each of the twelve 15-bit patterns in the library (see Figure 34).

A noise matrix (1×15) is created having 1 in the position where the input pixel was 1 and the corresponding library pixel was 0 (noise error). Similarly, a missing matrix (1×15) had 1 in the position where the input pixel was 0 and the corresponding library pixel was 1 (missing error).

Then, a total noise matrix (1×12) is created which corresponds to the sum of the noise errors for each pattern. Similarly, a total missing matrix (1×12) corresponds to the sum of the missing errors for each pattern.

The total noise matrix and the total missing matrix are added to create the total error matrix (1×12). Then the training vector generation algorithm finds the minimum in the total error matrix and outputs the corresponding pattern (number) if the corresponding total number of errors for that pattern in the noise matrix is less than one, and the corresponding total number of errors for that pattern in the missing matrix is less than two. It was observed that a noise pixel is more significant than a missing pixel, since it can lead to the recognition of an incorrect pattern. If there are two or more patterns (numbers) with the same minimum value in the total error matrix and also satisfy the total noise and missing matrix requirements, the training vector generation algorithm outputs the largest number.

If the input corresponds to a valid number, the input-target vectors for the training of the neural network are updated. Each target vector is a 10×1 column vector with 1 in the position of the number it represents, and 0’s elsewhere. If the incident pattern is zero, 1 appears as the 10^{th} element.

The above algorithm can be better illustrated with the use of the following example. Consider that the random 15-bit number representing the input is

[0 0 1 0 1 1 0 1 1 1 1 1 1 1 1]

which corresponds to the image in Figure 36.

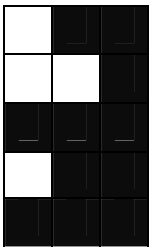


Figure 36. The input image for the example.

As stated above the 15-bit input number is XORed with each one of the vectors of the library (see Figure 34.). For example, consider XORing it with the vector corresponding to number 3 from the library, creating the noise and missing matrixes:

Input image:	[0 0 1 0 1 1 0 1 1 1 1 1 1 1 1]
Number 3 from the libray:	[1 0 1 0 1 1 0 1 0 1 1 1 1 1 1]
Noise matrix:	[0 0 0 0 0 0 0 0 1 0 0 0 0 0 0]
Missing matrix:	[1 0 0 0 0 0 0 0 0 0 0 0 0 0 0]

The total noise, missing and error matrixes obtained by comparing the image pattern to every number in the library are listed below. The values for number 3 are indicated in bold.

Total noise matrix:	[9	7	6	2	**1**	4	2	2	5	1	1	2]
Total missing matrix:	[3	1	0	2	**1**	2	2	3	1	3	2	3]
Total error matrix:	[12	8	6	4	**2**	6	4	5	6	4	3	5]

There are 12, 8 and 6 errors for the three possibilities of number one, 4 for numbers two, five and eight and so forth. The minimum value of the error matrix is of course 2 errors for number 3. Since the two requirements are met, the vector algorithm will output number three as the solution.

The training vector generation algorithm can also be extended to the recognition of the letters in the English alphabet, different shapes, etc. The library will not occupy a large amount of memory in the microprocessor, since the patterns are expressed in the form of binary numbers.

2. Training the Neural Network

The neural network receives the pattern of the object illuminated by light as a 15-element binary input vector. It is then required to identify the number by responding with a 10-element output vector. The neural network, thus, needs 15 neurons in the input layer and 10 neurons in the output layer. It is a two-layer log-sigmoid/ log-sigmoid network. The log-sigmoid transfer function was chosen because of its output range (0 to 1). The hidden layer has 15 neurons. The layout of the network of this feed-forward neural network is similar to that of Figure 23.

According to the above network architecture, the weight matrix of the hidden layer is a 15×15 matrix and the bias matrix is 15×1. Similarly, the weight matrix for the output layer is 10×15 and the corresponding bias vector is 10×1. Thus, we can express the network architecture in the form of matrix multiplications as follows:

$$y_{\text{hidden}} = \varphi(W_{\text{hidden}} \cdot x + b_{\text{hidden}}) \tag{4.1}$$

$$y_{\text{output}} = \varphi(W_{\text{output}} \cdot y_{\text{hidden}} + b_{\text{output}}) \tag{4.2}$$

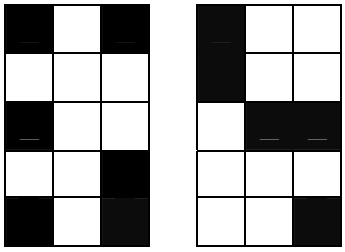
where φ is the log-sigmoid function and x is the input vector (15×1).

The network was trained using the training vectors obtained by the training vector generation algorithm (supervised learning) using MATLAB Neural Networks Toolbox [32] with a back propagation training function. Recall that after the completion of the network's training, the weights and bias values are fixed. At the completion of training, the weights and bias values were extracted by the network object that was created, and were implemented in the form of matrices in the Dynamic C algorithm for the BL2000. The code for Dynamic C algorithm is provided in Appendix B.

F. EXPERIMENTS

A series of experiments were performed that consisted of illumination of all the ten numbers of Figure 34 and also some noisy patterns (see Figure 26). The system in all cases correctly recognized the incident pattern. It also recognized numbers that the training vector generation algorithm in the Appendix A would characterize as nulls. For example, the images of Figure 37 were recognized as number five and number four,

respectively. The algorithm would characterize them as nulls, because the number of total errors for all the numbers in the library is greater than three. In addition, the noisy image of number five could also be recognized as number three, since they both have five missing errors. Recall that if there were more than one image that satisfied the requirements of the training vector generation algorithm, the algorithm selects the largest number.



No 5 No 4

Figure 37. Examples of noisy images that the neural network recognized.

In all the experiments the two robots approached the sensor according to the mapping described in Table 2. In addition, the position of the robots was changed and the experiments were conducted again. Again, the robots moved following the same mapping.

The following photographs (Figures 38 to 41) show the experiment conducted when number three was incident on the Si photodiode array. In this case, according to the mapping of Table 2, the yellow flashlight would switch ON forcing Robot 1 (on the right as seen in the figures) to approach the sensor.

Figure 38 shows the initial position of the two robots and number 3 incident on the photodiode array. The neural network algorithm has not been yet executed.

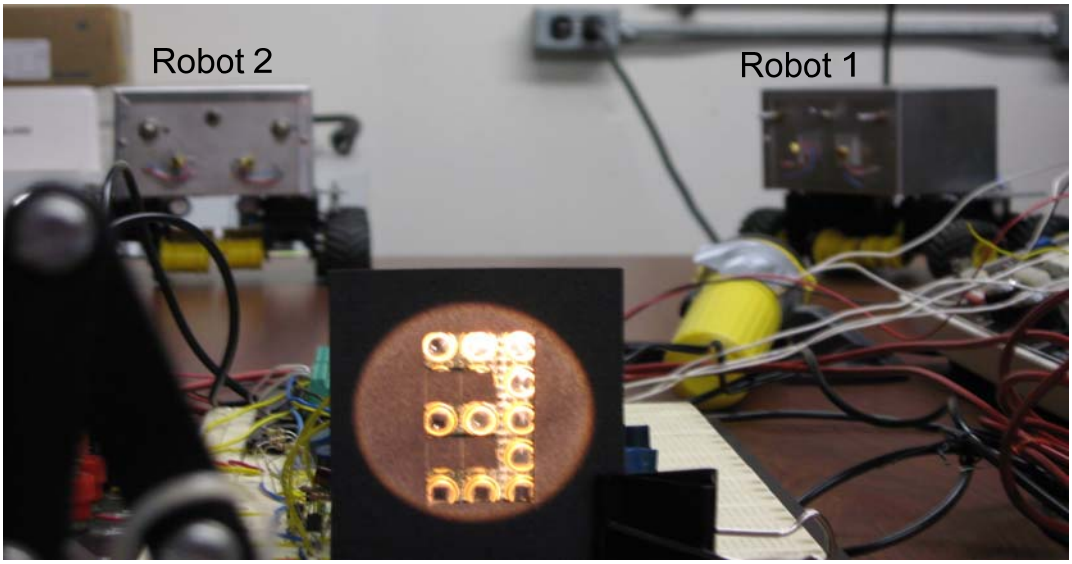


Figure 38. The initial position of the robots and number three incident on the Si photodiode array.

The neural network algorithm has recognized number 3 correctly and Figure 39 shows the yellow flashlight on the right switched ON. The light beam from the flashlight is incident on robot 1 (on the right).

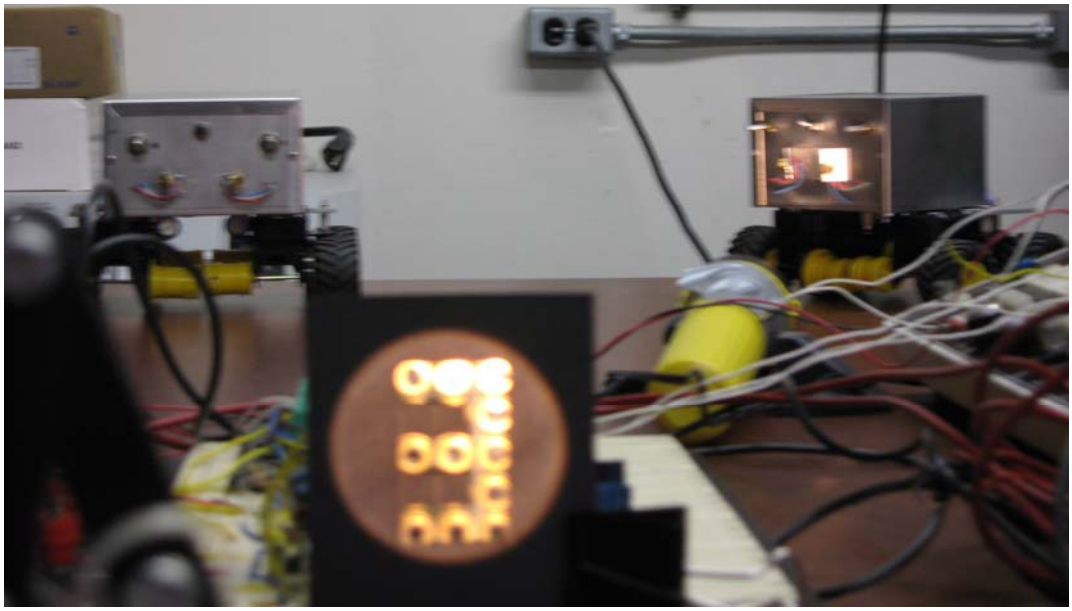


Figure 39. Light from the yellow flashlight is incident on robot 1 (on the right) after the correct recognition of number three.

Figure 40 shows the robot on the right responding to the light beam by starting to turn and approaching the source of light (yellow flashlight). The robot on the left is not moving.

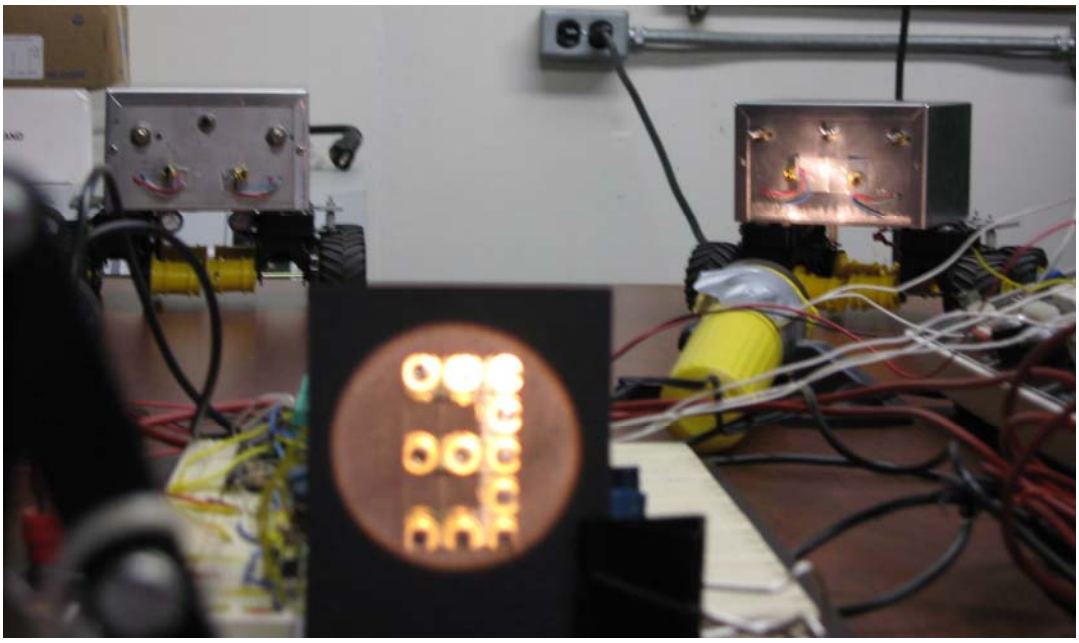


Figure 40. The robot on the right turns and follows the light beam. The robot on the left is not moving.

Figure 41 shows the robot aligned with the light beam. It continues to approach the sensor.

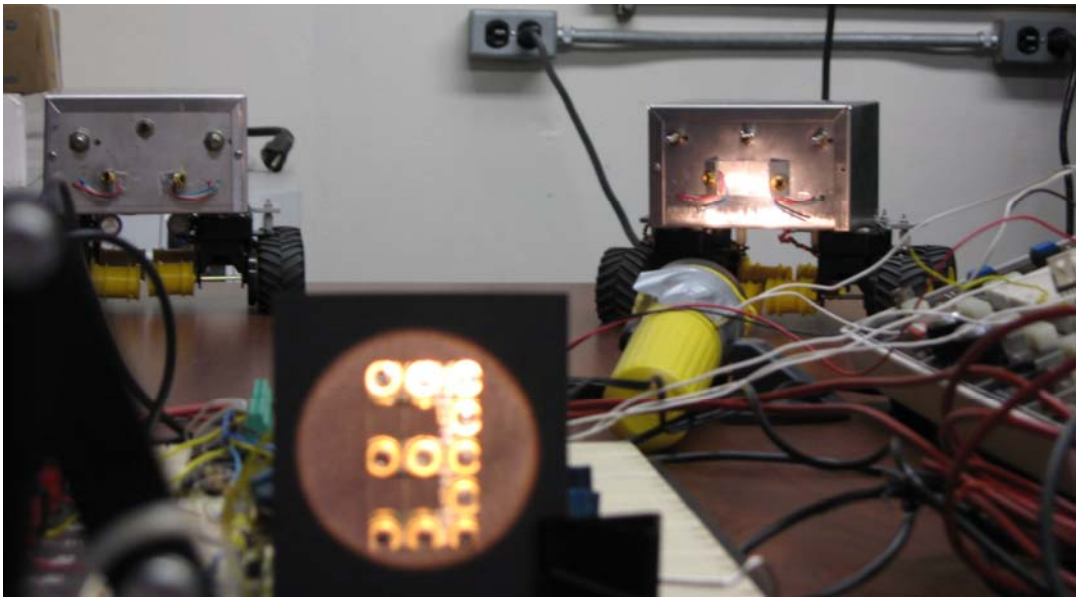


Figure 41. The robot on the right is aligned with the light beam from the yellow flashlight.

G. SUMMARY

In this chapter, an application of the unique characteristics of the pulsing circuit and neural networks was presented. A sensor that was able to recognize incident patterns (numbers 0 to 9) using a neural networks algorithm was presented. Two robots were activated by the sensing circuitry depending on the recognized number.

Since the photodiode array mimics the behavior of the retina neurons in the case of external stimuli, and the sensor recognized the incident pattern using the way humans do, it can be concluded that the constructed sensor models, to some extent, the human visual cortex.

THIS PAGE INTENTIONALLY LEFT BLANK

V. CONCLUSIONS

This thesis presented an in-depth investigation of the wavelength response of a photodiode-SCR pulse-mode circuit at room temperature. It was determined that the behavior of the circuit can be described by a renewal process statistical model. In addition, it was determined that at a low power of incident illumination, the pulse rate follows the photocurrent response of the photodiode, while at a higher power, saturation occurs and the pulse rate stays the same, independent of the wavelength of the incident light. A 2D sensor array that models, to some extent, the human visual cortex was constructed using the unique characteristics of the pulsing circuit and a neural networks algorithm. The sensor array was able to recognize incident patterns (numbers 0 to 9) and two robots were activated by the sensing circuitry depending on the recognized number.

A. SIGNIFICANT CONTRIBUTIONS

In Chapter II, it was shown that the pulse generating circuit can be used as a pulse-mode optical detector. The pulse-mode detection is not limited to Si photodiode's spectral range but can be extended to other wavelengths by replacing it with photodiodes made of other semiconductors, for example in the IR region.

In addition, the SCR circuit can be used as an optical detector without the use of the external Si photodiode but by directly shining light into the middle p-n junction of the SCR.

A back propagation algorithm based neural network was successfully trained to solve a real-world pattern recognition problem. The output of the neural network was used to activate selected robots emulating a form of networked communication.

B. FUTURE RESEARCH

Future research may focus on how to extract the spectral information (colors) of the incident illumination, probably by measuring the pulse rate for a constant power, and thus enabling the construction of colored pulse-mode optical sensors.

The SCR circuit has potential to be used as sensor elements for fabrication of an artificial vision system similar to that of the biological vision systems based on neurons and photoreceptors. Such sensor elements need to be relatively small and they should be constructed using CMOS integrated circuit fabrication technology.

APPENDIX A. THE TRAINING VECTOR GENERATION ALGORITHM

The algorithm produces the input and target vectors for the training of the neural network.

```
% CREATING THE RANDOM VECTORS
for k=1:3000
x=rand(1,15);
for i=1:15
    if x(i)<=0.5 x(i)=0;
    else x(i)=1;
    end
end

% THE NUMBERS LIBRARY (12x15)
numbers=[1 1 1 1 1 0 0 0 0 0 0 0 0 0 0; % 1
        0 0 0 0 0 1 1 1 1 1 0 0 0 0 0;
        0 0 0 0 0 0 0 0 0 0 1 1 1 1 1;
        1 0 1 1 1 1 0 1 0 1 1 1 1 0 1;  % 2
        1 0 1 0 1 1 0 1 0 1 1 1 1 1 1;  % 3
        1 1 1 0 0 0 0 1 0 0 1 1 1 1 1;  % 4
        1 1 1 0 1 1 0 1 0 1 1 0 1 1 1;  % 5
        1 1 1 1 1 1 0 1 0 1 1 0 1 1 1;  % 6
        1 0 0 0 0 1 0 0 0 0 1 1 1 1 1;  % 7
        1 1 1 1 1 1 0 1 0 1 1 1 1 1 1;  % 8
        1 1 1 0 1 1 0 1 0 1 1 1 1 1 1;  % 9
        1 1 1 1 1 1 0 0 0 1 1 1 1 1 1]; % 0

% THE TARGETS
targ=[1 0 0 0 0 0 0 0 0 0;
      0 1 0 0 0 0 0 0 0 0;
      0 0 1 0 0 0 0 0 0 0;
      0 0 0 1 0 0 0 0 0 0;
      0 0 0 0 1 0 0 0 0 0;
      0 0 0 0 0 1 0 0 0 0;
      0 0 0 0 0 0 1 0 0 0;
      0 0 0 0 0 0 0 1 0 0;
      0 0 0 0 0 0 0 0 1 0;
      0 0 0 0 0 0 0 0 0 1];

number=[1 1 1 2 3 4 5 6 7 8 9 0];         % The number matrix

% MISSING, NOISE AND ERROR MATRIXES
missing=zeros(12,15);
noise=zeros(12,15);
for j=1:12
    for i=1:15
        error(j,i)=xor(x(i),numbers(j,i));% XOR input and library
        if x(i)==0 && numbers(j,i)==1 missing(j,i)=1;
        else if x(i)==1 && numbers(j,i)==0 noise(j,i)=1;
            end
        end
```

```
        y(j)=length(find(error(j,:)));% Find how many 1s are in error
matrix
      end
end

%SELECTING IF THE RANDOM VECTOR IS A NUMBER OR NOT
output=zeros(1,12);
answer=zeros(1,12);
for j=1:12
    miss(j)=length(find(missing(j,:))); % Find the missing pixels
    noi(j)=length(find(noise(j,:))); % Find the noise pixels
    if y(j)==min(y) && noi(j)<=1 && miss(j)<=2
        output(j)=1;
        result=number(j);
        answer(j)=number(j);
    else if y(j)==min(y) && (noi(j)>1|| miss(j)>2)
        result=11; % Not a number; not to confuse with the actual 0
        end
    end
end

out=length(find(output)); %Select the maximum value if many
if out>1 ans=max(answer);
else ans=result;
end
tr(k,:)=[[x] ans]; % The total matrix of the 3000 random vectors and
their answer
end

% CREATING THE INPUT AND RANDOM VECTORS FOR THE NN TRAINING
l=0;
for k=1:3000
if tr(k,16)~=11 % If the answer is a number
   l=l+1;
   eis(l,:)=tr(k,1:15); % The input vector
   if tr(k,16)~=0
   ejod(l,:)=targ(tr(k,16),:); % The target vector
   else ejod(l,:)=[0 0 0 0 0 0 0 0 0 1]; % The target vector if the
answer is zero
end
end
end

eis'
ejod'
```

APPENDIX B. THE NEURAL NETWORKS ALGORITHM

The algorithm, implemented in the Wildcat BL2000 microprocessor, transforms the numerical character into a 15-bit binary number, executes the neural network algorithm for the recognition of the incident number, and outputs 4 V depending on the recognized number.

```
void main()
{
float whid[15][15]; // The weight matrix in the hidden layer
float bhid[15];       // The bias matrix in the hidden layer
float wo[10][15];   // The weight matrix in the output layer
float bo[10];         // The bias matrix in the output layer
int i,j,chan;         // Counter
int x[15] ;           // The 15-bit binary number
float sumh, sumo;
float y1[15], y12[15], y2[10], y22[10];
float outh[15], out[10]; // Outputs
int result;           // The recognized number

brdInit();

// TRANSFORMING THE IMAGE INTO A 15-BIT BINARY NUMBER
for (chan=0;chan<11;chan++){x[chan]=digIn(chan);}
for (j=16;j<20;j++){x[j-5]=digIn(j);}

// INITIALING THE MATRIXES
whid[0][0]=1.1151; whid[0][1]=-2.7015; whid[0][2]=1.6986; whid[0][3]=-1.944;
whid[0][4]=-0.24825; whid[0][5]=3.7892; whid[0][6]=0.75024; whid[0][7]=54.448;
whid[0][8]=1.0252; whid[0][9]=3.5385; whid[0][10]=1.1012; whid[0][11]=-3.3739;
whid[0][12]=-2.6153; whid[0][13]=-3.1354; whid[0][14]=-1.147;
whid[1][0]=1.6211; whid[1][1]=0.96897; whid[1][2]=-0.42898; whid[1][3]=2.5197;
whid[1][4]=3.6056; whid[1][5]=4.3128; whid[1][6]=2.3242; whid[1][7]=0.46121;
whid[1][8]=2.7742; whid[1][9]=2.1931; whid[1][10]=-0.99682; whid[1][11]=-1.3787;
whid[1][12]=0.55781; whid[1][13]=-0.6132; whid[1][14]=-1.1;
whid[2][0]=2.5123; whid[2][1]=1.2554; whid[2][2]=0.3446; whid[2][3]=0.44652;
whid[2][4]=0.0066263; whid[2][5]=2.4001; whid[2][6]=1.0062; whid[2][7]=1.1154;
whid[2][8]=0.87133; whid[2][9]=0.24408; whid[2][10]=0.96838; whid[2][11]=1.7319;
whid[2][12]=1.2804; whid[2][13]=0.69905; whid[2][14]=1.7121;
whid[3][0]=1.9014; whid[3][1]=-1.8316; whid[3][2]=-1.493; whid[3][3]=1.5801;
whid[3][4]=-1.7186; whid[3][5]=-0.4732; whid[3][6]=0.66505; whid[3][7]=-4.6199;
whid[3][8]=-0.28309; whid[3][9]=-1.9732; whid[3][10]=0.23392; whid[3][11]=-0.4127;
whid[3][12]=1.8527; whid[3][13]=-1.6917; whid[3][14]=0.3596;
```

whid[4][0]=-1.9892; whid[4][1]=9.9704; whid[4][2]=0.16548; whid[4][3]=1.761;
whid[4][4]=-1.8956; whid[4][5]=-2.5123; whid[4][6]=-1.735; whid[4][7]=-3.0296;
whid[4][8]=-1.6835; whid[4][9]=2.3727; whid[4][10]=0.43011; whid[4][11]=-2.6797;
whid[4][12]=-1.2235; whid[4][13]=6.258; whid[4][14]=1.5224;
whid[5][0]=2.1914; whid[5][1]=-0.38643; whid[5][2]=0.033233; whid[5][3]=2.3694;
whid[5][4]=-0.20546; whid[5][5]=-2.1494; whid[5][6]=1.2399; whid[5][7]=1.8566;
whid[5][8]=1.6122; whid[5][9]=-0.72319; whid[5][10]=-4.0497; whid[5][11]=46.55;
whid[5][12]=-3.5564; whid[5][13]=-0.71784; whid[5][14]=-0.60383;
whid[6][0]=8.8846; whid[6][1]=67.186; whid[6][2]=49.995; whid[6][3]=-0.97512;
whid[6][4]=50.683; whid[6][5]=111.69; whid[6][6]=-3.227; whid[6][7]=125.93;
whid[6][8]=63.503; whid[6][9]=-69.733; whid[6][10]=0.17626; whid[6][11]=1.6091;
whid[6][12]=-0.8442; whid[6][13]=-0.42252; whid[6][14]=1.5846;
whid[7][0]=-1.7783; whid[7][1]=-0.60334; whid[7][2]=-1.9669; whid[7][3]=-8.7291;
whid[7][4]=2.5132; whid[7][5]=0.079422; whid[7][6]=-0.37376; whid[7][7]=-2.8243;
whid[7][8]=-1.0768; whid[7][9]=0.96501; whid[7][10]=2.1383; whid[7][11]=4.2494;
whid[7][12]=0.53943; whid[7][13]=1.8538; whid[7][14]=1.1508;
whid[8][0]=-1.315; whid[8][1]=2.5408; whid[8][2]=4.1961; whid[8][3]=0.40434;
whid[8][4]=5.3455; whid[8][5]=-3.9133; whid[8][6]=1.7885; whid[8][7]=2.1151;
whid[8][8]=0.06177; whid[8][9]=1.0234; whid[8][10]=0.13352; whid[8][11]=-1.0564;
whid[8][12]=1.0598; whid[8][13]=-2.7309; whid[8][14]=-1.5575;
whid[9][0]=-0.53735; whid[9][1]=3.7698; whid[9][2]=1.2955; whid[9][3]=-4.6255;
whid[9][4]=-1.4838; whid[9][5]=-1.017; whid[9][6]=-0.67876; whid[9][7]=1.7354;
whid[9][8]=-0.60686; whid[9][9]=-0.7437; whid[9][10]=0.32551; whid[9][11]=-2.0388;
whid[9][12]=-0.46289; whid[9][13]=1.3596; whid[9][14]=0.09237;
whid[10][0]=-0.048471; whid[10][1]=-0.54528; whid[10][2]=0.66144;
whid[10][3]=-2.3585; whid[10][4]=-0.93147; whid[10][5]=5.9086; whid[10][6]=4.637;
whid[10][7]=-3.0118; whid[10][8]=-0.42583; whid[10][9]=-1.3797;
whid[10][10]=-1.5196; whid[10][11]=-3.4068; whid[10][12]=-2.0638;
whid[10][13]=5.596; whid[10][14]=3.2566;
whid[11][0]=-1.7715; whid[11][1]=-1.7161; whid[11][2]=0.38348; whid[11][3]=7.9012;
whid[11][4]=-1.0152; whid[11][5]=-4.4804; whid[11][6]=-1.0867; whid[11][7]=2.0493;
whid[11][8]=-1.2941; whid[11][9]=-2.5384; whid[11][10]=-1.3753;
whid[11][11]=1.2197; whid[11][12]=0.50018; whid[11][13]=1.5849;
whid[11][14]=1.0266;
whid[12][0]=11.756; whid[12][1]=9.8158; whid[12][2]=5.9646; whid[12][3]=8.3252;
whid[12][4]=11.462; whid[12][5]=7.9797; whid[12][6]=6.1047; whid[12][7]=6.8338;
whid[12][8]=7.581; whid[12][9]=8.3535; whid[12][10]=11.65; whid[12][11]=9.4582;
whid[12][12]=11.253; whid[12][13]=7.5912; whid[12][14]=8.666;
whid[13][0]=-4.017; whid[13][1]=-2.5472; whid[13][2]=-1.6971; whid[13][3]=2.9153;
whid[13][4]=0.1464; whid[13][5]=1.8255; whid[13][6]=0.10638; whid[13][7]=1.9214;
whid[13][8]=-1.1747; whid[13][9]=-2.7951; whid[13][10]=-2.1675;
whid[13][11]=-1.1079; whid[13][12]=-3.5356; whid[13][13]=-0.92155;
whid[13][14]=0.62997;
whid[14][0]=-0.85232; whid[14][1]=9.3823; whid[14][2]=-1.0032; whid[14][3]=-3.0234;
whid[14][4]=-1.658; whid[14][5]=-1.8727; whid[14][6]=-0.80714; whid[14][7]=0.63568;

whid[14][8]=-0.94508; whid[14][9]=-1.2905; whid[14][10]=-0.50503;
whid[14][11]=7.269; whid[14][12]=-0.82886; whid[14][13]= -0.19963;
whid[14][14]=0.17969;

bhid[0]=0.053344; bhid[1]=-6.8579; bhid[2]=-7.8418; bhid[3]=1.1753;
bhid[4]=-0.50417; bhid[5]=-2.9422; bhid[6]=-7.9094; bhid[7]=-1.2096;
bhid[8]=-0.5453; bhid[9]=-0.55113; bhid[10]=2.506; bhid[11]=0.76469;
bhid[12]=12.777; bhid[13]=2.128; bhid[14]=-5.2282;

wo[0][0]=-3.1499; wo[0][1]=14.295; wo[0][2]=-38.991; wo[0][3]=1.3126;
wo[0][4]=-1.41; wo[0][5]=2.4433; wo[0][6]=0.036097; wo[0][7]=0.59583;
wo[0][8]=2.9094; wo[0][9]=2.7127; wo[0][10]=-0.02006; wo[0][11]=2.2386;
wo[0][12]=0.85595; wo[0][13]=2.6322; wo[0][14]=3.4302;
wo[1][0]=1.2106; wo[1][1]=-0.34645; wo[1][2]=7.8339; wo[1][3]=-2.7135;
wo[1][4]=-29.104; wo[1][5]=0.97414; wo[1][6]=-2.82; wo[1][7]=-8.2588;
wo[1][8]=1.9552; wo[1][9]=-35.17; wo[1][10]=-2.3387; wo[1][11]=-1.235;
wo[1][12]=-1.4838; wo[1][13]=-2.4491; wo[1][14]=-24.488;
wo[2][0]=2.0818; wo[2][1]=-3.0169; wo[2][2]=0.19541; wo[2][3]=-78.754;
wo[2][4]=4.6759; wo[2][5]=3.2048; wo[2][6]=0.57963; wo[2][7]=10.907;
wo[2][8]=-5.751; wo[2][9]=-15.798; wo[2][10]=1.5818; wo[2][11]=0.44159;
wo[2][12]=0.14435; wo[2][13]=-14.343; wo[2][14]=-36.306;
wo[3][0]=-1.9307; wo[3][1]=-50.31; wo[3][2]=7.2547; wo[3][3]=-9.652;
wo[3][4]=-0.90293; wo[3][5]=0.91575; wo[3][6]=-0.62483; wo[3][7]=-6.8554;
wo[3][8]=6.6231; wo[3][9]=2.1126; wo[3][10]=-1.0943; wo[3][11]=1.727;
wo[3][12]=-2.3299; wo[3][13]=-45.238; wo[3][14]=0.96692;
wo[4][0]=2.6644; wo[4][1]=0.25638; wo[4][2]=2.1165; wo[4][3]=-40.07;
wo[4][4]=0.5855; wo[4][5]=-40.5; wo[4][6]=-2.0873; wo[4][7]=18.132;
wo[4][8]=-0.023695; wo[4][9]=9.285; wo[4][10]=-2.4227; wo[4][11]=-13.615;
wo[4][12]=-1.7007; wo[4][13]=-40.67; wo[4][14]=-3.3003;
wo[5][0]=0.7028; wo[5][1]=4.0522; wo[5][2]=0.42236; wo[5][3]=-15.698;
wo[5][4]=15.235; wo[5][5]=-28; wo[5][6]=-1.9001; wo[5][7]=-28.37;
wo[5][8]=-0.66381; wo[5][9]=-3.2593; wo[5][10]=-3.6703; wo[5][11]=1.9334;
wo[5][12]=-0.61421; wo[5][13]=-14.879; wo[5][14]=-27.82;
wo[6][0]=-1.4828; wo[6][1]=-19.534; wo[6][2]=11.565; wo[6][3]=5.0781;
wo[6][4]= 0.95207; wo[6][5]=-1.1225; wo[6][6]=6.5078; wo[6][7]=1.4821;
wo[6][8]=-28.454; wo[6][9]=-2.1797; wo[6][10]=-2.8221; wo[6][11]=-5.3879;
wo[6][12]=-0.98611; wo[6][13]=-1.3695; wo[6][14]=-3.2758;
wo[7][0]=0.87892; wo[7][1]=0.96952; wo[7][2]=1.0071; wo[7][3]=-80.43;
wo[7][4]=-1.878; wo[7][5]= 2.6771; wo[7][6]=-2.6747; wo[7][7]=-7.2728;
wo[7][8]=0.54515; wo[7][9]=-28.723; wo[7][10]=2.0612; wo[7][11]=1.0262;
wo[7][12]=-1.6747; wo[7][13]=-80.67; wo[7][14]=12.464;
wo[8][0]=-4.194; wo[8][1]=1.3118; wo[8][2]=-0.49264; wo[8][3]=-80.29;
wo[8][4]=-0.62697; wo[8][5]=-0.7391; wo[8][6]=-2.2352; wo[8][7]=1.328;
wo[8][8]=0.29036; wo[8][9]=-1.338; wo[8][10]=-1.1556; wod[8][11]=-29.144;
wo[8][12]=0.87643; wo[8][13]=-80.6; wo[8][14]=19.817;
wo[9][0]=-35.374; wo[9][1]=11.048; wo[9][2]=2.5404; wo[9][3]=-3.25;

```
wo[9][4]=2.716; wo[9][5]=-0.30584; wo[9][6]=1.8626; wo[9][7]=-1.4096;
wo[9][8]=-1.8317; wo[9][9]=-53.26; wo[9][10]=2.4139; wo[9][11]=0.69022;
wo[9][12]=-0.36229; wo[9][13]=-60.04; wo[9][14]=7.1564;

bo[0]=4.3172; bo[1]=4.2399; bo[2]=-2.7749; bo[3]=4.2172; bo[4]=1.0485;
bo[5]=2.7045; bo[6]=0.20849; bo[7]=-1.0732; bo[8]=-0.72449; bo[9]=-5.7185;
// HIDDEN LAYER
sumh=0;
for(i=0;i<15;i++)
  {
  for(j=0;j<15;j++)
  {  sumh=sumh+whid[i][j]*x[j];}
  y1[i]=sumh;
  sumh=0;}

for(i=0;i<15;i++) { y12[i]=y1[i]+bhid[i];}
for(i=0;i<15;i++) { outh[i]=1/(1+exp(-y12[i]));} // Sigmoid

// OUTPUT LAYER
sumo=0;
for(i=0;i<10;i++)
  {
  for(j=0;j<15;j++)
  {  sumo=sumo+wo[i][j]*outh[j];}
  y2[i]=sumo;
  sumo=0;}

for(i=0;i<10;i++) { y22[i]=y2[i]+bo[i];}
for(i=0;i<10;i++) { out[i]=1/(1+exp(-y22[i]));}   // Sigmoid

// THE RECOGNIZED NUMBER
for (j=0;j<10;j++)
  {
  if (out[j]>0.98)
    {result=j+1;
     if (result==10){result=0;}
    }
  }
printf("%d\t",result);

// VOLTAGE OUTPUT DEPENDING ON THE RECOGNIZED NUMBER
if (result==1 || result==2 || result==3) {anaOutVolts(0,4.0);}
if (result==4 || result==5 || result==6) {anaOutVolts(1,4.0);}
if (result==7 || result==8 || result==9) {anaOutVolts(0,4.0); anaOutVolts(1,4.0);}
}
```

LIST OF REFERENCES

[1] G. Karunasiri, "Spontaneous Pulse Generation Using Silicon Controlled Rectifier," *Applied Physics Letters*, Vol. 89, pp.1-3, 2006.

[2] S. M. Sze, *Physics of Semiconductor Devices 2nd Edition*, John Wiley & Sons, New York, 1981.

[3] D. A. Moore, "Optical Detection Using Four-Layer Semiconductor Structures," Master's Thesis, Naval Postgraduate School, Monterey, California, 2005.

[4] A. P. Matos, "Characterization and Application of Four-Layer Semiconductor Structures in Pulse Mode Generation," Master's Thesis, Naval Postgraduate School, Monterey, California, 2006.

[5] V. A. K. Temple, "Comparison of light triggered and electrically triggered thyristor turn-on," *IEEE Trans. Electron Devices*, Vol. ED-28, No. 7, pp. 860-865, 1981.

[6] E. Margalit, M. Maia, J. W. Weiland, R. J. Greenberg, G. Y. Fujii, G. Torres, D. V. Piyathaisere, T. M. O'Hearn, W. Liu, G. Lazzi, G. Dagnelie, D. A. Scribner, E. deJuan, and M. S. Humayun, "Retinal prosthesis for the blind," *Survey of Ophthalmology*, Vol. 47, Issue 4, pp 335-356, 2002.

[7] M. S. Humayun, E. de Juan, J. D. Weiland, G. Dagnelie, S. Katona, R. Greenberg, and S. Suzuki, "Pattern electrical stimulation of the human retina," *Vision Research*, Vol. 39, Issue 15, pp 2569-2576, 1999.

[8] D. Yanai, J. D. Weiland, M. Mahadevappa, G. Y. Fujii, E. de Juan, R. J. Greenberg, R. Williamson, V. Cimmarusti, and M. S. Humayun, "Visual perception in blind subjects with microelectronic retinal prosthesis," *Investigative Ophthalmology & Visual Science*, Vol. 44, p U702, 2003.

[9] M. Mahadevappa, J. D. Weiland, D. Yanai, I. Fine, R. J. Greenberg, and , M. S. Humayun, "Perceptual thresholds and electrode impedance in three retinal prosthesis subjects," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, Vol. 13, Issue 2, p 201, 2005.

[10] A. Y. Chow, V. Y. Chow, K. H. Packo, J. S. Pollack, G. A. Peyman, and R. Schuchard, "The artificial silicon retina microchip for the treatment of vision loss from retinitis pigmentosa," *Archives of Ophthalmology*, Vol. 122, Issue 4, pp 460-469, 2004.

[11] T. Bensaoula, C. A. Garcia, A. Zomorrodian, N. J. Wu, and A. Ignatiev, “Thin film optical detectors for retinal implant-A "bionic" eye,” *Investigative Ophthalmology & Visual Science*, Vol. 42, Issue 4, p S814, 2001.

[12] R. A. Normann, E. M. Maynard, K. S. Guillory, and D. J. Warren, “Cortical implants for the blind,” *IEEE Spectrum*, Vol. 33, pp 54-&, 1996.

[13] M. B. Schubert, A. Hierzenberger, H. J. Lehner, and J. H. Werner, “Optimizing photodiode arrays for the use as retinal implants,” *Sensors & Actuators A-Physical*, Vol. 74, Issue 1-3, pp 193-197, 1999.

[14] D. Ziegler, P. Linderholm, M. Mazza, S. Ferazzutti, D. Bertrand, A. M. Ionescu and Ph. Renaud, “An active microphotodiode array of oscillating pixels for retinal stimulation,” *Sensors & Actuators A-Physical*, Vol. 110, Issue 1-3, pp 11-17, 2004.

[15] W. H. Dobelle, “Artificial vision for the blind by connecting a television camera to the visual cortex,” *Asaio Journal*, Vol. 46, Issue 1, pp 3-9, 2000.

[16] R. A. Normann, D. J. Warren, J. Ammermuller, E. Fernandez, and I. Perlman, “Light adaptation and sensitivity controlling mechanisms in vertebrate photoreceptors,” *Progress in Retinal and Eye Research*, Vol. 17, Issue 4, pp 523-563, 1998.

[17] R. A. Normann, D. J. Warren, J. Ammermuller, E. Fernandez, and S. Guillory, “High-resolution spatio-temporal mapping of visual pathways using multi-electrode arrays,” *Vision Research*, Vol. 41, Issue 10-11, pp 1261-1275, 2001.

[18] P. Troyk, T. Bak, J. Berg, D. Bradley, S. Cogan, R. Erickson, C. Kufta, D. McCreery, E. Schmidt, and T. Towle, “A model for intracortical visual prosthesis research,” *Artificial Organs*, Vol. 27, Issue 11, pp 1005-1015, 2003.

[19] E. L. Dereniak and G. D. Boreman, *Infrared Detectors and Systems 1st Edition*, John Wiley & Sons, New York, 1996.

[20] Website: “THOR LABS, Inc,” http://www.thorlabs.com, last accessed 11/04/08.

[21] C. Therrien and M. Tummala, *Probability for Electrical and Computer Engineers*, CRC Press, 2004.

[22] M. J. Berry and M. Meister, “Refractoriness and Neural Precision,” *Journal of Neuroscience*, Vol. 18, Issue 6, pp. 2200-2211, 1998.

[23] MATLAB, “exppdf” (Statistics Toolbox),” MATLAB 7.1 Software, The Math Works, Inc, 1994-2005.

[24] D. Heeger, “Poisson Model of Spike Generation,” 05 Sep 2000, http://www.cns.nyu.edu/~david/handouts/poisson.pdf, last accessed 11/04/08.

[25] F. Mainardi, R. Gorenflo and A. Vivoli, “Beyond the Poisson Renewal Process: A Tutorial Survey,” *Journal of Computational and Applied Mathematics,* Vol. 205, pp. 725-735, 2007.

[26] MATLAB, “gampdf” (Statistics Toolbox),” MATLAB 7.1 Software, The Math Works, Inc, 1994-2005.

[27] S. Haykin, *Neural Networks, a comprehensive foundation*, Prentice Hall, 1999.

[28] J. Freeman and D. Skapura, *Neural Networks: Algorithms, Applications and Programming Techniques*, Addison-Wesley Publishing Company, Inc, 1991.

[29] S. Deutsch and A. Deutsch, *Understanding the Nervous System: An Engineering Perspective*, IEEE Press, NJ, 1993.

[30] P. Scherz, *Practical Electronics for Inventors 2nd edition*, McGraw-Hill, 2007.

[31] User’s Manual, “Wildcat BL2000 C-Programmable Single-Board Computer with Ethernet,” Z-WORLD, Inc. 2001-2004.

[32] MATLAB, “Neural Networks Toolbox,” MATLAB 7.1 Software, The Math Works, Inc, 1994-2005.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Professor Gamani Karunasiri, Code PH/Kg
   Naval Postgraduate School
   Monterey, California

4. Professor Murali Tummala, Code ECE/Tm
   Naval Postgraduate School
   Monterey, California